ATARI® 130XE™

PERSONAL COMPUTER

Manuale d'Uso

INFORMAZIONI IMPORTANTI

Come per qualsiasi apprecchiatura elettrica, il Computer ATARI 130XE usa e produce energia e radiofrequenze. Se non viene installato secondo le istruzioni di questo manuale, il computer potrebbe causare delle interferenze alla vostra radio o TV.

Se avete motivo di credere che il computer causi delle interferenze alla vostra radio o TV, provate a spegnerlo e riaccenderlo. Se le interferenze cessano quando il computer è spento, allora è probabile che sia il computer a causare le interferenze.
Con il computer acceso potreste rimediare al problema con uno dei seguenti accorgimenti:

- Orientare in modo diverso l'antenna radio o TV.
- Spostare il computer in un'altra posizione rispetto a quella della radio o della televisione.
- Alimentare il computer da un'altra presa di corrente in modo che il computer e la radio o la televisione siano su circuiti differenti.

Se necessario consultate il vostro rivenditore od un tecnico radio e televisione specializzato.

ATTENZIONE: Questa apparecchiatura è certificata per conformità alle limitazioni della Classe B che riguarda i dispositivi di calcolo elettronici secondo la Sottosezione I, Parte 15 del regolamento FCC. Questo regolamento prevede una ragionevole protezione contro tali interferenze quando l'apparecchiatura è usata in ambienti residenziali. Però non garantisce che l'interferenza non avvenga in particolari case o residenze. Solo periferiche (dispositivi input/output, terminali, stampanti etc) certificate per la conformità alle limitazioni della Classe B possono essere collegate a questo computer.

L'uso di periferiche non certificate con questo computer possono causare interferenze alla radio o TV.

Ogni sforzo è stato fatto per assicurare la precisione delle informazioni sul prodotto contenute in questo manuale. Poiché stiamo costantemente migliorando e aggiornando il software e l'hardware del computer. L'Atari Corp. non può garantire la correttezza del materiale pubblicato dopo la data di pubblicazione e non si assume alcuna responsabilità per cambi, errori ed omissioni.

IL GIUSTO COMPUTER ALLA PORTATA DI TUTTI

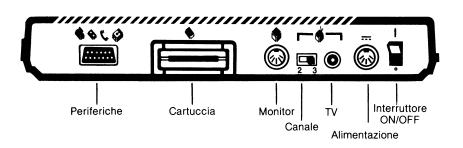
Benvenuti nel mondo dei personal computer. L'ATARI 130XE, pur appartenendo al gruppo dei piccoli computer, ha una grande potenza e versatilità.
Il 130XE è il piú nuovo tra i computer di questa serie: con i suoi 131,072 Bytes RAM, il 130XE possiede una memoria di capacità due volte superiore al computer ATARI 65XE. E grazie a questa caratteristica il 130XE ha una potenza ed un numero di possibilità notevolmente superiori. Potrete inserire un maggior numero di dati, sia nel word processor che nell'archivio dati e le funzioni di tipo educativo saranno molto piú avanzate.
Oltre ad avere maggiori capacità di memoria, il 130XE ha una tastiera lineare, capacità grafiche e sonore superiori, disponibilità di migliaia di programmi ed una vasta gamma di accessori compatibili.
L'ATARI 130XE è il computer piú indicato per attività gestionali, educative e di semplice divertimento. Ed inoltre il 130XE ha il linguaggio ATARI BASIC incorporato, che vi permette di iniziare immediatamente a scrivere i vostri programmi.
Questo manuale vi fornisce le informazioni di cui avete bisogno nel modo piú semplice possibile.
La parte 1 vi spiega il modo in cui disimballare e poi collegare il vostro computer ATARI 130XE, controllando se funziona perfettamente; vi spiega come usare la tastiera, caricare le cartucce di software ed allargare il sistema aggiungendo delle unità periferiche.
La parte 2 è una lezione introduttiva alla programmazione in Basic. L'ATARI 130XE ha molteplici applicazioni che potrete sfruttare integralmente appena avrete ben compreso le caratteristiche del linguaggio Basic in esso incorporato.
Se conoscete già la programmazione, potete passare alla parte 3 di questo manuale, nella quale troverete alcuni esempi di programmi ed il materiale sufficiente ad illustrarvi le possibilità di programmazione di questo computer.
I programmatori piú esperti potranno inoltre trovare tutte le informazioni necessarie per accedere integralmente alle RAM del 130XE.
Non appena avrete installato il vostro ATARI 130XE e lo avrete messo in funzione, leggete la sezione "Risorse" di questo manuale.

Indice

PARTE 1
PRIMI PASSI

Tastiera

Tastiera
Di linea funzionale con 62 tasti incluso il tasto Help e 4 tasti con funzioni speciali.

 Connettore periferiche

Connette un cavo seriale I/O alle unità periferiche, come unità a dischi, registratori, stampanti e modem.

Allocamento cartuccia

Permette l'uso di cartucce per un caricamento semplice e veloce di programmi.

Espansione Interfaccia potenziata per cartuccia (ECI)

Connette periferiche ad alta velocità come le unità a disco molto veloci.

 Connettore per monitor

Consente l'uso di un monitor sia a colori che in bianco e nero

 Selettore canale TV

Seleziona il 2° o il 3° canale per una immagine migliore

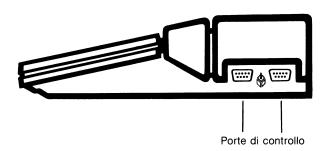
 Connettore TV

Spina per il cavo della scatola di commutazione TV

⎓ Spina di alimentazione

Collega il computer alla rete di alimentazione

I/O Interruttore ON/OFF

Accende e spegne il computer

 Porte di controllo

Permette di collegare tavolette grafiche, comandi a leva e a cloche, comandi numerici. Usate la porta n. 1 se utilizzate un solo comando.

Fare i collegamenti

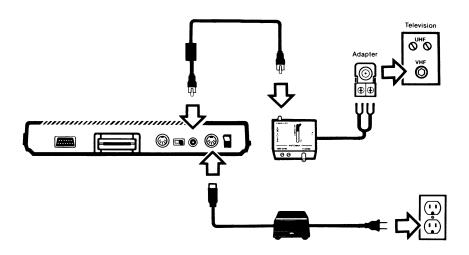
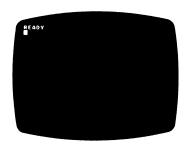
Quando aprite la scatola d'imballo del vostro ATARI 130XE assicuratevi che contenga quanto segue:

- Computer ATARI
- Alimentatore corrente alternata
- Manuale utente del 130XE
- Scheda registrazione della garanzia

Se il corredo non è completo contattare il vostro rivenditore (la garanzia copre i prodotti mancanti).
Sarebbe una buona idea conservare la scatola d'imballo nel caso che vogliate trasportare o spedire il vostro computer.

Collegare il vostro computer ATARI è facilissimo.
Tutto quello che vi serve è un cacciavite ed un televisore; l'Atari Corp. vi fornisce il resto. Il vostro ATARI 130XE può essere collegato ad un normale televisore in bianco e nero oppure a colori, o potete acquistare un video monitor per computer da usare esclusivamente come schermo di visualizzazione.

Per collegare il computer al monitor, seguite le istruzioni fornite con il monitor.
Avrete bisogno di un cavo (abitualmente fornito con il monitor) per collegare il monitor al connettore del monitor sul retro del vostro ATARI 130XE.

COLLEGAMENTO DEL COMPUTER ALLA SCATOLA DI COMMUTAZIONE TV

1. Con l'interruttore ON-OFF del computer sulla posizione OFF, inserite il connettore rotondo dell'alimentatore in corrente alternata nel connettore di alimentazione sul vostro computer. Inserite l'altro capo nella presa di corrente.
2. Accendete il vostro televisore e selezionate il canale 2 o 3 a seconda di quale dei due si riceve meno bene nella vostra zona. Predisporre l'interruttore di canale sul retro del vostro computer sul medesimo canale. Accendete il computer: dopo pochi secondi vedrete la parola READY.

L'indicatore READY vi informa che il BASIC ATARI è pronto per le vostre istruzioni e che il vostro computer è collegato correttamente. Se l'indicatore READY non appare sul vostro televisore, provate a sincronizzare l'immagine sul vostro televisore.

(Per informazioni piú dettagliate vedere Appendice E: Ricerca Guasti).

Quando lasciate il computer acceso per piú di nove minuti senza usare la tastiera, il computer incomincerà a cambiare i colori per proteggere il vostro televisore da bruciature di colori. Quando spegnete il vostro computer, aspettate circa cinque secondi prima di riaccenderlo.

CURA DEL VOSTRO ATARI 130XE

- Togliete la polvere con un panno senza filacce e leggermente umido.
- Non usate solventi o detergenti per pulire il computer.
- Mantenete i liquidi distanti dalla vostra area di lavoro.
- Evitare di fumare vicino al computer.
- Non muovete il computer piú di quanto sia necessario.

Fare i tests

Al momento dell'accensione il vostro ATARI 130XE esegue automaticamente dei tests. Tuttavia voi potete eseguire tre tests separati; questo per assicurarvi che il computer funzioni bene. Si raccomanda di eseguire tutti e tre i tests dopo che è stato assiemato per la prima volta. Per eseguire i tests accendete il computer mentre tenete abbassato il tasto OPTION. Potete pure accedere ai programmi di test battendo sulla tastiera la parola BYE dopo che il BASIC ATARI ha dato l'indicazione READY. (Se avete una unità a disco collegato al vostro computer, assicuratevi che sia spento quando eseguite il SELF TEST).

Il menu SELF TEST comparirà sullo schermo.
Come il menu di un ristorante, questo menu offre una gamma di scelta: MEMORY TEST, AUDIOVISUAL TEST, KEYBOARD TEST e ALL TEST. Premete il tasto [Select] per scegliere un test, poi premete il tasto [Start] . Premete [Help] per ritornare al menu di SELF TEST. Quando selezionate ALL TEST il computer esegue tutti e tre i tests. I tests vengono ripetuti fino a che premete il tasto [Help] .
Premete [Reset] per fermare i test e ritornare il BASIC ATARI.

TEST DI MEMORIA

Questo test controlla la ROM (Read Only Memory) e la RAM (Random Access Memory). La ROM è una memoria che non si può cancellare e che contiene il sistema operativo del computer. La RAM è una memoria che viene usata dai programmi che usate o scrivete voi stessi.

Il test di memoria funziona un po' come un semaforo. Due barre appaiono sullo schermo quando la ROM viene provata. Se le barre diventano verdi la ROM va bene e potete procedere, se diventano rosse dovete fermarvi poiché la ROM è guasta.

Durante il test RAM, vengono provati 112K di RAM. I restanti 16K RAM sono disponibili solo con alcuni programmi.
Durante il test, vengono provati 48 piccoli quadrati (ognuno dei quali rappresenta 1K RAM) e 4 rettangoli piú grandi (ognuno dei quali rappresenta 16K RAM).
Se un quadrato diventa bianco e poi verde, la sezione corrispondente funziona. Se un quadrato diventa rosso, ciò significa che la RAM relativa non funziona.

Note: se accedete al test RAM utilizzando l'ATARI BASIC digitando BYE, otterrete 40 e non 48 piccoli quadrati, poiché l'ATARI BASIC utilizza per sé 8K RAM.

TEST AUDIO VISIVO

Il test audio visivo prova le quattro voci sonore programmabili e le capacità grafiche. Una chiave di sol e rigo musicale compariranno sullo schermo sopra il numero della voce in prova. Sei note sono suonate e visualizzate sul vostro schermo. Le sei note si ripetono per ciascuna delle quattro voci.

Se il numero di una voce appare sullo schermo ma non sentite la musica, quella voce non funziona bene. Assicurarsi che il volume del vostro televisore sia adeguatamente regolato.

Se state usando un monitor, assicuratevi che il cavo audio del monitor sia fissato saldamente.
I colori visualizzati sullo schermo devono essere consistenti durante ogni prova.

TEST DI TASTIERA

Quando selezionate questo test, lavorate con due tastiere: quella sotto le vostre dita e quella visualizzata sullo schermo. Quando premete un tasto, il tasto corrispondente sullo schermo lampeggerà in forma video inversa (carattere blu su fondo bianco) e si sentirà una nota. La barra di spaziatura e i tasti di funzioni speciali sono già in forma video inversa cosicché battendo uno di quelli cambia il tasto corrispondente sullo schermo in forma video normale. I tasti SHIFT e CONTROL lampeggiano solo se premuti simultaneamente ad un altro tasto.

Due tasti non lampeggiano o suonano e fanno terminare il test quando vengono premuti; sono il tasto Help, che vi riporta al menu SELF TEST e il tasto Reset, che vi riporta al BASIC.

Il tasto Break non funziona in SELF TEST e non richiama nessuna funzione se premuto. Se un tasto, eccetto il tasto Break, non lampeggia sullo schermo quando viene premuto, significa che non funziona.

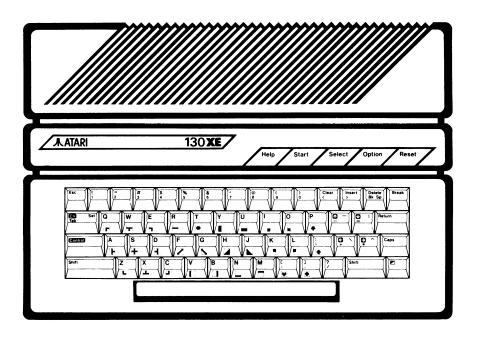
In aggiunta al SELF TEST opzionale il vostro ATARI 130XE esegue automaticamente il Test di Memoria ogni qualvolta il computer viene acceso. Se i circuiti del computer non superano i test, lo schermo indicherà MEMORY TEST con i rettangoli verdi e rossi. Fintanto che non avete problemi con il computer, i SELF TEST possono essere eseguiti molto occasionalmente.

CARICAMENTO DELLE CARTUCCE DI SOFTWARE

Dopo aver eseguito i tests sul vostro ATARI 130XE, potete caricare una cartuccia di software. Per caricare un programma di software, assicuratevi che il computer sia spento, prendete la cartuccia in modo che l'etichetta sia in alto, inseritela nella porta per cartucce sul retro del computer e accendete il computer. Se avete già usato un'altra cartuccia, spegnete il computer, togliete la cartuccia ed inserite la nuova cartuccia. Prima di riaccendere il computer, aspettate qualche secondo.

Se non volete usare il BASIC ATARI (il linguaggio di programmazione incorporato nel vostro ATARI 130XE) lo potete scartare temporaneamente in diversi modi: tenete premuto il tasto [Option] quando accendete il computer; battete la parola BYE per uscire dal SELF TEST; oppure, se il computer è collegato ad una unità a disco, battete la parola DOS per attivare il Sistema Operativo del Disco. (Per altri particolari, vedere Scoprire il BASIC ATARI).

CAPIRE LA VOSTRA TASTIERA

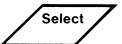
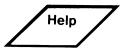
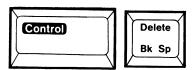
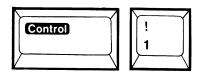
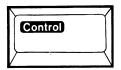
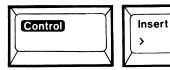
Sebbene la tastiera dell'ATARI 130XE sia simile e funzionale come quella di una macchina da scrivere, ci sono dei tasti particolari che vi consentono di usare degli accorgimenti speciali.
Alcuni tasti come [Shift] e [Control] sono usati in combinazione con altri. Queste combinazioni a doppio tasto consentono l'esecuzione di comandi e funzioni aggiuntive senza aumentare le dimensioni della tastiera.

Le seguenti descrizioni generali spiegano le funzioni dei tasti speciali e come vengono usati. Poiché le funzioni di un tasto possono cambiare da un programma all'altro, è importante consultare il manuale del programma che state usando. Per una descrizione particolare di come funziona la tastiera con il BASIC ATARI incorporato, vedete il capitolo: "Aggiornare con la tastiera dell'ATARI 130XE" nella Parte 2: Programmare con il BASIC ATARI.

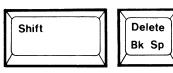
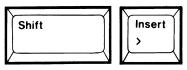
Arresta il computer nel mezzo di un'attività e fa ritornare il programma alla posizione iniziale. Premendo [Reset] si produce lo stesso effetto che accendere e spegnere il computer, con due eccezioni. La prima è che nella maggior parte dei programmi il [Reset] non cancella la memoria di RAM, mentre spegnendo ed eccendendo il computer, la memoria viene cancellata. La seconda è che l'uso del tasto [Reset] risparmia l'usura dell'alimentatore e dell'interruttore ON/OFF.

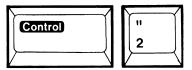
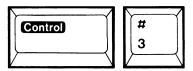
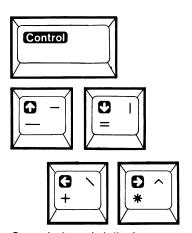
Temporaneamente scarta il BASIC se premuto quando il computer viene acceso. In alcuni programmi seleziona delle varianti all'interno del programma.

Solitamente interrompe qualsiasi funzione che il computer stia svolgendo. Fare riferimento alle istruzioni specifiche di ogni programma.

Varia da programma a programma ma di solito viene usato per passare da un menu all'altro

Normalmente indica al computer di iniziare un gioco od un programma. Consultare il manuale di software.

Spesso usato per selezionare una delle parecchie applicazioni all'interno di un programma. Poiché la sua funzione varia, consultare il manuale di software.

In alcuni programmi vi dà delle istruzioni supplementari per cavarvi d'impaccio.

Cancella il carattere sotto il cursore e sposta i rimanenti caratteri sulla linea per chiudere lo spazio lasciato vuoto.

Nella maggior parte dei linguaggi di programmazione arresta una visualizzazione mobile quando volete vedere una lista istruzioni.

Sempre usato assieme ad un altro tasto. Stampa degli speciali caratteri grafici quando usato con i tasti alfebetici con il BASIC ATARI.

Inserisce uno spazio nel BASIC ATARI, sebbene il suo uso specifico varia da programma a programma.

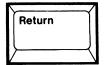
Nella maggior parte dei programmi, BASIC ATARI incluso, cancella il carattere alla sinistra del cursore e sposta il cursore uno spazio a sinistra. Non chiude lo spazio lasciato vuoto dalla cancellatura.

Cancella una linea di programma dallo schermo.
L'istruzione di programma rimane nella memoria del computer.

Inserisce una linea vuota nel BASIC ATARI.

Assieme ad altri tasti scrive lettere maiuscole lasciando indisturbate le lettere minuscole.

Suona il cicalino (Il volume della TV deve essere alzato per sentirlo).

Produce un segnale di fine archivio (EOF = end-of-file) per un programma che prende i dati dalla tastiera (Usato dai programmatori esperti).

Quando i tasti delle frecce vengono usati con il tasto Control le frecce muovono il cursore in alto, in basso, a sinistra e a destra.

Blocca il computer nella condizione di lettere maiuscole. Dovete
sempre premere ancora il tasto Shift per battere i caratteri nella parte superiore dei tasti numerici e simbolici.

Sposta il computer dalle maiuscole alle minuscole nel BASIC ATARI ed altri programmi. Usato nel BASIC ATARI per uscire dalla condizione Control Lock (caratteri grafici).

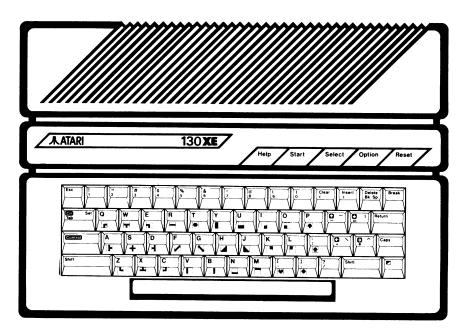
Blocca il computer nella condizione Control. Viene usato quando si
battono una serie di comandi che richiedono l'uso del testo Control, come nel caso di creazione di grafici con il BASIC ATARI.

Molti tasti del 130XE si ripeteranno automaticamente se li tenete premuti per piú di mezzo secondo.

Dice al computer che voi avete scritto o redatto una linea di programma. Fa ritornare il cursore al margine sinistro.

Accende e spegne le condizioni di video inverso.

Presentazione della grafica ATARI

La tastiera dell'ATARI 130XE ha incorporati 29 caratteri grafici. Questi possono esser usati per rendere piú interessante un grafico o per eseguire lavori artistici. Per rappresentare i caratteri grafici sul vostro schermo, premete il tasto [Control] ed un tasto qualsiasi mostrato qui sotto. Se avete intenzione di usare piú caratteri, sarà piú conveniente bloccare il computer nella condizione di [Control] premendo [Control] e [Caps]. Premete il tasto [Caps] per uscire dalla condizione di blocco e ritornare ai caratteri alfabetici.

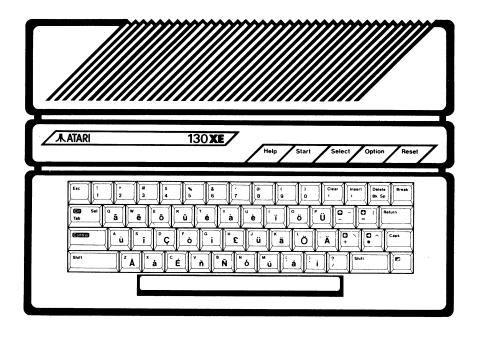
Aggiunta di una Tastiera Internazionale

Normalmente quando scrivete a macchina in una lingua straniera dovete inserire a mano i vari accenti, cediglie, ed altri segni diacritici. Con il computer ATARI 130XE questa fatica vi viene risparmiata poiché è dotato di una serie di caratteri internazionali.

I caratteri internazionali sono disponibili quando usate il BASIC ATARI. Il diagramma qui sotto mostra i caratteri internazionali che sono associati ai tasti delle lettere. Per attivare i caratteri internazionali, battete la frase qui sotto e premete il tasto [Return] .

```
POKE 756,204
```

Premendo il tasto [Control] e qualsiasi tasto mostrato qui sotto, verrà prodotto un carattere internazionale invece di uno grafico. Per ripristinare la condizione di caratteri grafici e l'uso normale della tastiera, battete la linea qui sotto e premere il tasto [Return]

```
POKE 756,224
```

Scoprire il BASIC ATARI

Il vostro computer ATARI 130XE ha incorporato il BASIC ATARI, uno dei piú diffusi linguaggi di programmazione. Il BASIC (Beginner's Allpurpose Symbolic Instruction Code) è stato sviluppato al Dartmonth College negli anni 60 per insegnare la programmazione dei computer agli studenti del collegio.
Da allora è diventato uno dei piú conosciuti linguaggi di programmazione per i computers.

Sebbene il BASIC sia un unico linguaggio, ne esistono varie versioni leggermente differenti, come il BASIC ATARI che ha delle importanti e singolari caratteristiche. Per esempio, alcune parole particolari del BASIC ATARI generano suoni e creano dei grafici a colori. In aggiunta, il BASIC ATARI è stato progettato appositamente per il programmatore inesperto.
A differenza di altre versioni del BASIC, il BASIC ATARI controlla ogni linea di programma che avete scritto e vi dirà se avete fatto un errore di grammatica o sintassi. Naturalmente quando avete imparato a programmare con il BASIC ATARI, vi sarà piú facile imparare le altre versioni del BASIC.

Il BASIC ATARI è immediatamente disponibile quando accendete il computer. Se non volete usare il BASIC, tenete abbassato il tasto [Option] quando accendete il computer. Un altro modo per uscire dal BASIC è di battere sulla tastiera la parola BYE, che inizia il SELF TEST, oppure DOS che attiva il sistema operativo del disco (solo quando il computer è collegato ad un'unità a disco).

La seconda parte di questo manuale, Programmare con il BASIC ATARI, contiene un breve corso d'istruzione per il programmatore inesperto. A differenza degli altri corsi, che vi fanno subito studiare la sintassi del linguaggio, questo vi insegna subito a scrivere programmi, fare giochi di parole, risolvere problemi matematici e usare le possibilità grafiche sonore dell'ATARI 130XE. Quando avete terminato questo corso, capirete le varie funzioni dei tasti e come adoperare il computer. Il corso vi permette di sfruttare al massimo il vostro ATARI 130XE.

Se sapete già programmare in BASIC, le appendici vi daranno la maggior parte delle informazioni di cui avrete bisogno per lavorare a vostro piacimento. L'Appendice A contiene programmi dimostrativi a vari livelli di difficoltà per farvi fare un po' di pratica prima di incominciare a scrivere i vostri programmi in BASIC ATARI.

L'Appendice B, contiene parole riservate al BASIC, una lista di comandi usati nel BASIC ATARI ed una breve descrizione di quello che ciascun comando ordina al computer di fare. Per i programmatori esperti l'Appendice C, Serie di caratteri ATASCII, elenca le posizioni decimali e esadecimali dei caratteri in codice ATASCII.
Inoltre, i programmatori che desiderano conoscere integralmente le capacità di memoria del 130XE, possono accedervi tramite l'appendice H: Accedere alla potenzialità del 130XE, che tramite la funzione POKE, vi fornisce il controllo delle locazioni di memoria. Quando un messaggio d'errore appare sullo schermo guardate l'appendice D: Messaggi di Errore, per vedere che cosa non ha funzionato. Vedere l'Appendice F: Risorse, per trovare i nomi di altre fonti d'informazione per il BASIC ATARI.

Uso del Software

Il vostro ATARI 130XE ed il vostro televisore sono le parti hardware del vostro sistema. Il software si riferisce ai programmi del computer che dicono al vostro ATARI 130XE come fare e come farlo.
Alcuni programmi di software sono incorporati nel computer, altri li potete acquistare ed altri ancora li potete scrivere voi stessi.

SOFTWARE INCORPORATO

Il software incorporato è permanente, codificato nel chip della ROM (memoria di sola lettura) nel computer. Il vostro ATARI 130XE contiene una chip di ROM codificata, con il linguaggio di programmazione ATARI BASIC ed il programma del Sistema Operativo.
Nell'ATARI 130XE, il Sistema Operativo contiene programmi che permettono alla tastiera, al monitor, al registratore di programmi e alla maggior parte delle stampanti ATARI di comunicare tra loro.

SOFTWARE COMMERCIALE

I programmi commerciali di software sono scritti dalla ATARI Corp. e da programmatori professionisti e li potete acquistare dai rivenditori. Videogiochi, fogli elettronici, eleborazione di testi e lezioni di matematica, per citarne alcuni, sono tutti programmi di software che sono disponibili per intrattenimento e per scopi didattici e gestionali.

Il computer ATARI 130XE è perfettamente compatibile con il 65XE e l'800XL. Potete scegliere qualsiasi programma che volete usare purché siano stati progettati per essere usati sui computer ATARI 130XE e ATARI 800XL.
Il software e l'hardware dell'800XL possono essere usati sul 130XE.

Il vostro ATARI 130XE è dotato per l'uso delle cartucce di software. Se volete usare il software su dischi o nastri magnetici, dovete acquistare una unità a disco o un registratore di programmi.

SOFTWARE SCRITTO DALL'UTENTE

Una volta che avete imparato il linguaggio di programmazione, siete in condizione di scrivere i vostri programmi. L'ATARI 130XE viene fornito con il linguaggio di programmazione BASIC ATARI incorporato. Il corso nella Parte 2 di questo manuale vi insegna come usare questo linguaggio per scrivere un programma in Basic. Tuttavia, potete anche scrivere programmi in altri linguaggi, come in LOGO ATARI, ad esempio, inserendo la cartuccia contenente il linguaggio di vostra scelta.

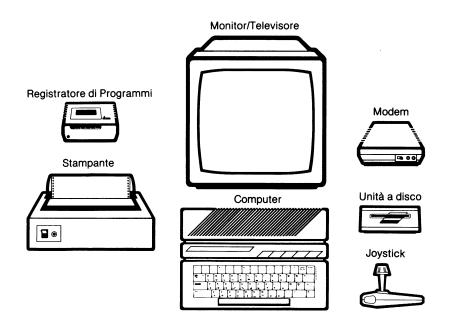
COME CONSERVARE IL VOSTRO LAVORO

Se scrivete i vostri programmi, certamente li vorrete conservare per usarli nuovamente. Se usate il vostro computer per elaborare testi, probabilmente vorrete conservare anche il lavoro che avete fatto. Spegnendo il computer tutto il vostro lavoro viene perso ma, se collegate al vostro computer una unità a disco od un registratore di programmi, potete memorizzare il vostro lavoro per usarlo piú tardi. L'unità a disco conserva l'informazione su dischi, mentre il registratore di programmi usa le normali musicassette. Le istruzioni fornite con l'unità a disco ed il registratore di programmi vi diranno come usare questi dispositivi (Per ulteriori informazioni vedere: Espandere il vostro sistema).

COME CARICARE LE CARTUCCE DI SOFTWARE

Caricare le cartucce commerciali di software pronte per l'uso nel vostro ATARI 130XE è semplicissimo. La cartuccia di software va inserita nell'alloccamento per cartucce sul retro del vostro computer. Prima di tutto, assicurarsi che il computer sia spento. Inserire o togliere una cartuccia mentre il computer è acceso vuol dire danneggiare sia il computer che la cartuccia. Prendete la cartuccia con l'etichetta in su. Inserite saldamente la cartuccia nella apposita porta e accendere il computer. Se avete appena usato un'altra cartuccia, aspettate qualche secondo prima di accendere di nuovo il computer.

Espansione del Vostro Sistema

Il vostro ATARI 130XE è il cuore di un potente e versatile sistema ed ha una periferica che vi assiste sia per un lavoro che per un gioco. Le periferiche piú comuni sono illustrate qui sotto:

TV O MONITOR

Il televisore od il monitor sono gli schermi di visualizzazione del vostro computer. Parecchie persone preferiscono usare il monitor perché ottengono un'immagine piú precisa e non interferisce con l'uso del televisore. Si può usare sia monitor o TV a colori che bianco-nero. Tuttavia parecchi programmi sono stati fatti per mostrare la gamma variopinta della grafica a colori del computer ATARI 130XE.

UNITÀ A DISCO

L'unità a disco ATARI 1050 permette di conservare e richiamare i programmi ed ogni altra informazione che create con il vostro computer. Dato che molti programmi sono ottenibili su dischi, l'unità a disco vi consente di accedere ad una vostra libreria di programmi già pronti nelle aree della elaborazione testi, gestione finanziaria, didattica ed intrattenimento.

REGISTRATORE DI PROGRAMMI

Il registratore di programmi ATARI 1010 è un dispositivo molto economico per conservare informazioni. Come una unità a disco, vi permette di conservare i programmi e qualsiasi altra informazione. Sebbene non sia efficiente come una unità a disco per uso professionale o gestionale, ha tuttavia a sua disposizione una libreria di programmi già pronta per usi didattici e di intrattenimento.

STAMPANTE

Come proprietario di un ATARI avete un ampia gamma di stampanti tra le quali scegliere. I modelli XTM201 e XTC201 sono stampanti non-impact a matrice disponibili per l'ATARI 130XE. Entrambe possono stampare grafici ed il modello XTC201 stampa a colori. Per stampare testi e lettere, grafici e programmi, è consigliato l'uso della XMM801, stampante di alta qualità e molto economica. Ed infine, per stampare lettere d'affari che richiedano un'impostazione molto professionale, potete usare la stampante mod. XDM121, letter quality con caratteri a margherita.

MODEM

Il modem ATARI XM301, il vostro computer ed un telefono vi danno accesso ad una rete mondiale di comunicazioni elettroniche. Se avete un amico con un modem collegato al computer potete scrivervi lettere. Se siete collegati ad una delle tante banche dati che sono disponibili, potete avere accesso ai tanti gruppi utenti, posta elettronica, notiziari e materiali informativi.

JOYSTICK

I joystick sono dispositivi molto diffusi per programmi didattici e di intrattenimento. Con il joystick potete controllare il vostro computer senza toccare la tastiera.

Parte 2
Programmare
con il BASIC ATARI

Aggiornare con la Tastiera ATARI 130XE

Non importa quanto siate bravi nello scrivere a macchina, dovete sempre familiarizzare con le caratteristiche speciali della tastiera dell'ATARI 130XE prima di cominciare ad aggiornare con il BASIC ATARI.

FUNZIONE DI RIPETIZIONE AUTOMATICA

Cominciate con lo scrivere la lettera A:

```
A
```

Tenete premuto il tasto [A] e notate come appaiono le file di A. Quando la linea è completa, il cursore automaticamente inizia la fila seguente. Non c'è bisogno di premere il tasto [Return] .

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

State usando la funzione di ripetizione automatica della tastiera ATARI 130XE. La maggior parte dei tasti, incluso la barra spaziatrice, hanno la ripetizione automatica. Avete sentito il cicalino quando la terza línea di A era quasi completa? Quello era il cicalino di avvertimento, una funzione incorporata nel BASIC ATARI: indica che la linea di istruzione sta diventando troppo lunga. Una linea di istruzione non può essere piú lunga di tre linee.

MESSAGGI D'ERRORE

Premete il tasto [Return] . Dovreste vedere la parola ERROR sullo schermo seguita dalle tre linee di A che avete scritto. Il vostro computer sta comunicando con voi. Vi sta dicendo che non capisce cosa state scrivendo perché le file di A non sono parte del linguaggio BASIC. Cancellate lo schermo premendo [Return] fino a che il Messaggio d'Errore scompare. Per evitare di avere dei messaggi d'errore mentre state seguendo le istruzioni di redazione, non premere il tasto [Return] fino a che ricevete l'ordine preciso di premerlo.

MAIUSCOLE E MINUSCOLE

Per scrivere l'A minuscola premere il tasto [Caps] una volta e tener premuto il tasto [A] . Sullo schermo dovreste vedere:

```
aaaaaaaa
```

Per tornare alle lettere maiuscole, premere il tasto [Caps] e scrivere ancora delle A. Sullo schermo dovreste vedere:

```
AAAAAAA
```

Provate a scrivere una parola – una parola che inizia con A, come ATARI. Scrivete le seguenti parole passando da lettere maiuscole e lettere minuscole con il tasto [Caps]

```
ATARI130XE atari130xe
```

I numeri sono sempre gli stessi, sia che scriviate maiuscolo o minuscolo. A differenza di una macchina per scrivere, il computer ha dei tasti per controllare le maiuscole e gli spostamenti. In ambedue i casi, minuscole e maiuscole, il simbolo che viene scritto è sempre quello mostrato nella parte inferiore del tasto. Per avere il simbolo sulla parte superiore del tasto, premete il tasto [Shift] . Sulla tastiera ci sono due tasti [Shift] . Potete usare sia l'uno che l'altro.

Usando i tasti [Caps] [Shift] e [1] provate a scrivere cosí:

```
!!! ATARI130XE !!! !!! atari130xe !!!
```

Fate delle prove con parole differenti, lettere e segni di punteggiatura.

SIMBOLI GRAFICI

Parecchi tasti hanno due o tre simboli. Ogni tasto con lettere ha una lettera nella parte in alto ed un simbolo grafico sul frontale. Altri tasti invece hanno tre simboli o parole, tutti sulla parte alta del tasto. La funzione di un tasto è attivata premendo il tasto soltanto, un'altra funzione premendo [Shift] ed il tasto e la terza premendo [Control] ed il tasto. I simboli grafici sono prodotti premendo [Control] ed il tasto.

Per scrivere un simbolo grafico (il simbolo sul frontale di un tasto con lettere) usate il tasto [Control] sulla sinistra della tastiera. Prima premete il tasto [Control] . Mentre tenete premuto il tasto [Control] , premete il tasto con il simbolo grafico voluto. Rilasciate i due tasti contemporaneamente.

Premete il tasto [Control] e provate a scrivere ATARI 130XE
Dovreste ottenere questo:

Sullo schermo ci saranno solo sette caratteri. I tasti numerici non hanno simboli grafici. Se usate il tasto [Control] quando premete un tasto numerico, nessun simbolo grafico verrà rappresentato.

I simboli grafici sono molto utili per fare cornici, ricami e lavori artistici in genere. Potete bloccare la tastiera nella posizione grafica premendo [Control] e [Caps] . Premendo [Caps] solo una volta si ritornerà alla condizione di lettere minuscole.

CONTROLLO DEL CURSORE

Il tasto [Control] è frequentemente usato per guidare lo spostamento del cursore. Il cursore è il quadratino bianco che indica la vostra posizione sullo schermo. Individuate il tasto freccia verso l'Alto [- -] vicino alla lettera P.
La freccia, come il testo [Control] , è marcata in bianco. Questa marcatura bianca indica che la funzione della freccia è attivata soltanto dal tasto [Control] . Premete il tasto [Control] e poi il tasto Freccia verso l'Alto [- -] e notate come il cursore si muove verso l'alto dello schermo. Quando arriva in alto il cursore ritorna in basso allo schermo e incomincia a muoversi in alto di nuovo. Adesso provate ad usare altri tasti direzionali [= |], [* ^], [+ \]. Ricordatevi di usare il tasto [Control] .

CANCELLARE LO SCHERMO

Il tasto [Control] viene usato spesso con il tasto [Clear] per cancellare qualsiasi cosa sullo schermo. Tenete premuto il tasto [Control] mentre premete il tasto [Clear] . Questa azione dovrebbe cancellare lo schermo e spostare il cursore in alto a sinistra dello schermo. Provate di nuovo.

Adesso riempite lo schermo con altre lettere, numeri, parole e simboli grafici. Questa volta usate il tasto [Shift] con il tasto [Clear] per cancellare lo schermo.
Ambedue [Shift] [Clear] e [Control] [Clear] puliscono lo schermo e spostano il cursore in alto a sinistra.

INSERIRE

Il tasto [Control] viene usato con il tasto [Insert] per inserire degli spazi in una linea.
Per sperimentare questa funzione scrivete:

```
!!! ATARI130XE !!!
```

Spostate il cursore sopra la prima A di ATARI. Tenendo premuto il tasto [Control], premete il tasto [Insert] 11 volte. Ecco che cosa succede:

```
!!!           ATARI130XE !!!
```

Undici spazi sono stati aggiunti in mezzo alla linea.
Questa funzione è molto utile quando bisogna inserire delle parole. Usando i tasti di controllo di cursore (le frecce), spostate il cursore nello spazio subito dopo il terzo punto esclamativo. Premete la barra spaziatrice una volta e scrivete nello spazio vuoto come mostrato qui sotto:

```
!!! THIS IS AN ATARI130XE !!!
```

Per aggiungere linee vuote, piuttosto che spazi vuoti, tenete premuto il tasto [Shift], poi premete il tasto [Insert]. Una linea vuota apparirà sullo schermo. Provate ad inserire altre linee vuote, ma non cosí tante da non avere niente sullo schermo. Lasciate un po' di spazio per l'esercizio seguente.

ELIMINARE

Usate e il tasto [Control] con il tasto [Delete Bk Sp].
Eliminare è tanto facile quanto inserire. Spostate il cursore sulla lettera T della parola THIS. Mentre tenete premuto il tasto [Control], premete il tasto [Delete Bk Sp] per 11 volte. Ecco quello che vedrete:

```
!!! ATARI130XE !!!
```

Adesso sapete come funzionano i tasti [Control] e [Delete Bk Sp]. Per scoprire che cosa fa il tasto [Delete Bk Sp] quando è premuto da solo, spostate il cursore sulla prima A e poi premere [Delete Bk Sp] quattro volte. Ecco quello che vedrete sullo schermo:

```
ATARI130XE !!!
```

Quando viene usato da solo, il tasto [Delete Bk Sp] sposta il cursore a sinistra, cancellando mentre si sposta ma senza riempire gli spazi. Usando il tasto [Control] con il tasto [Delete Bk Sp] si eliminano i caratteri alla destra e si chiudono gli spazi lasciati vuoti.

La terza funzione del tasto [Delete Bk Sp] richiede l'uso del tasto [Shift]. Tenendo premuto il tasto [Shift] mentre si preme [Delete Bk Sp] si elimina una linea intera e il cursore si sposta sul margine sinistro. Non importa dove nella linea si trovi il cursore; quando premete [Shift] [Delete Bk Sp] la linea intera è eliminata e non è piú recuperabile.

TABULATORE

Sullo schermo pulito, spostate il cursore sul margine sinistro e scrivete un asterisco. Premete il tasto [Tab]. Ogni qualvolta il cursore si ferma, scrivete un asterisco. Dovreste ottenere sei asterischi distanziati sullo schermo come mostrato qui sotto:

Premete solo il tasto [Tab] e osservate come si ferma sempre sulle stesse posizione di tabulazione preselezionate.
Il primo tabulatore è a cinque spazi dal margine sinistro (il normale capoverso di paragrafo) e i seguenti tabulatori sono distanziati di otto spazi. Spostate il cursore sopra il primo asterisco e muovetelo di tre spazi. Tenete premuto [Shift] e poi premete [Tab] per attivare la funzione [Set Tab]. Spostate il cursore indietro sul margine sinistro, poi premete il tasto [Tab]. Il cursore salta alla nuova posizione di tabulazione.

Continuate a premere il tasto [Tab]. Si sposterà su tutte le posizioni di tabulazione preselezionate.
Quando il cursore scende sulla linea seguente, ignora la nuova posizione di tabulazione (ma su tutte le linee seguenti il cursore andrà su tutte le posizioni di tabulazione – le nuove e le preselezionate). Muovete il cursore sul primo asterisco e premete [Tab]. La nuova posizione di tabulazione si trova ancora lí.

Spostate il cursore indietro sul margine sinistro. Premete [Tab] per muoverlo nella prima posizione di tabulazione (tre spazi in dentro). Usate il tasto [Control] con il tasto [Tab] per attivare la funzione [Clr Tab]. Premete il tasto [Tab] per andare alla prossima posizione di tabulazione ed eliminarla. Spostate il cursore indietro sul margine sinistro della stessa linea e premete solo il tasto [Tab]. Il cursore dovrebbe saltare due posizioni di tabulazione. Continuate a premere il tasto [Tab] fino a che il cursore si sposta sulla linea successiva. [Clr Tab] non ha cancellato la seconda posizione di tabulazione di questa linea (tuttavia ambedue le posizioni di tabulazione sono state cancellate nelle linee successive).

VIDEO INVERSO

Scrivete la parola ATARI. Localizzate il tasto Video Inverso ◪ e premetelo solo una volta. Scrivete ATARI di nuovo. Premete il tasto video inverso e scrivere la parola ATARI. Ecco cosa apparirà sul vostro schermo:

ATARI ATARI ATARI

Il video inverso o negativo scrive lettere blu su fondo bianco, il contrario dei normali colori di schermo. Questa funzione è molto utile quando volete mettere in rilievo le lettere dei vostri programmi. Basta un tocco del tasto Video Inverso per cambiare il modo in cui le lettere vengono rappresentate.

TASTI VARI

Un altro tasto importante è il tasto Escape [Esc]. Quando lo premete una volta non succede niente. Quando lo premete piú di due volte appare sullo schermo questo grafico ␛. Premete [Return] e provate di nuovo. Nei capitoli seguenti avrete bisogno di usare il tasto [Esc].

Il tasto [Break] si trova nell'angolo in alto a destra. Quando premete questo tasto, il cursore va al margine sinistro della linea successiva. Imparerete a usarlo nel capitolo sul looping.

Premendo ciascuno dei seguenti tasti [Help] [Start] [Select] [Option] non succederà niente. Questi tasti sono programmabili e hanno spesso delle funzioni particolari nei programmi di software.

Dopo aver premuto il tasto [Reset], lo schermo si svuota per un secondo o due e la parola Ready appare. Il tasto [Reset] fa ripartire il sistema. Dovete usare questo tasto con molta parsimonia perché in molti programmi fa perdere l'informazione che avete inserito o state inserendo nel computer.

Scrivere un Semplice Programma in BASIC

NEW, LIST, PRINT, RUN

Una volta che avete imparato ad usare la tastiera, è facile scrivere il vostro primo programma. Per incominciare, cancellate lo schermo ed assicuratevi che il cursore sia sul margine sinistro.

NEW: CANCELLA LA MEMORIA DEL COMPUTER

Scrivete la parola NEW, poi premete [Return] :

```
NEW
```

NEW prepara il computer per una nuova serie di istruzioni cancellando qualsiasi istruzione che possa trovarsi nella memoria.

LIST: CONTROLLA LA MEMORIA DEL COMPUTER

Per essere sicuro che non ci sia niente nella memoria del computer, chiedete al computer di visualizzare qualsiasi istruzione che possa contenere. Scrivere LIST da solo su una linea, poi premete [Return] :

```
LIST
```

Se avete scritto correttamente la parola NEW non ci dovrebbe essere niente sul vostro schermo. Potete quindi cominciare un nuovo programma. Battete la prima linea di istruzione nel computer. Battete la linea esattamente come appare qui sotto, poi premete [Return] dopo le ultime virgolette:

```
10 PRINT "I HEARD OF A POET NAMED SAM"
```

Tutte le linee di istruzione in BASIC sono numerate. Quando battete questo programma di una sola linea, assicuratevi che l'1 e lo 0 nel numero 10 siano numeri e non lettere. Se usate lettere invece di numeri, avrete un Messaggio d'Errore.

In un programma una linea d'istruzione numerata non può essere piú lunga di una riga sullo schermo. Quando il cursore è al termine della riga, automaticamente va alla riga successiva. Dovrete premere [Return] solo alla fine di una linea d'istruzione per dire al computer che avete terminato di scrivere l'istruzione e che dovrebbe memorizzarla. Quando premete [Return] , non succede niente di grave: il cursore si sposta sul margine sinistro, cosí che potete scrivere un'altra linea di programma.

RUN: ESEGUE L'ISTRUZIONE

Per far si che il computer esegua il vostro programma, dovete battere RUN. Il comando RUN dice al computer che deve eseguire le istruzioni ricevute. Battete RUN, poi premete [Return] per vedere che cosa succede:

```
RUN
I HEARD OF A POET NAMED SAM
```

La prima ed unica istruzione del computer, linea 10, era di stampare la frase tra le virgolette. Cancellate lo schermo, battete di nuovo RUN e premete [Return]. Il computer esegue l'istruzione di nuovo e visualizza I HEARD OF A POET NAMED SAM.

Sebbene l'istruzione non si trovi piú sullo schermo, il computer si ricorda di quello che deve fare. Il vostro programma è memorizzato nella RAM (Memoria di accesso causale) che è la parte programmabile nella memoria del computer. Quando battete LIST, il computer mostra sullo schermo tutte le istruzioni memorizzate nella parte RAM della sua memoria. Battete LIST. Ecco che cosa deve apparire sul vostro schermo:

```
LIST
10 PRINT "I HEARD OF A POET NAMED SAM"
```

Se lo schermo non riporta quanto sopra, può darsi che vi siate dimenticati di premere [Return] alla fine di ogni istruzione o battere LIST su di una riga da solo.
Battete la linea qui sotto, poi date il comando RUN:

```
20 PRINT "I MET HIM ONE DAY, AND TO MY DISMAY,"
RUN
```

Le parole tra le virgolette di ambedue le righe del programma appaiono sullo schermo. Battete LIST per vedere l'istruzione che il computer ha memorizzato nella RAM. Appariranno cosí ambedue le righe 10 e 20.

NUMERAZIONE DELLE LINEE: CREARE UN ORDINE

Ogni linea d'istruzione nel BASIC deve iniziare con un numero. I numeri si chiamano "numeri di riga". Il computer esegue le istruzioni incominciando dal numero piú piccolo e continuando il programma fino a che tutte le istruzioni sono state completate.

Il metodo abituale è di numerare le linee in multipli di dieci cosicché tra un'istruzione e la seguente ci sono abbastanza numeri disponibili

per inserire delle altre istruzioni, se necessario. Ora provate ad aggiungere una linea. Aggiungete la linea 15 (mostrata nella pagina seguente) ed istruite il computer a girare il programma. Il vostro schermo dovrebbe apparire come segue:

```
15 PRINT "WHOSE POEMS WERE THE TALK OF THE LAND."
RUN
I HEARD OF A POET NAMED SAM
WHOSE POEMS WERE THE TALK OF THE LAND.
I MET HIM ONE DAY, AND TO MY DISMAY,
```

Il computer ha automaticamente inserito la riga 15 tra le righe 10 e 20. Scrivete un'altra linea.

```
30 PRINT "HIS BRAINS WERE SILICON-SAND."
RUN
LIST
```

I comandi RUN e LIST fanno si che le quattro linee dell'istruzione PRINT appaiano sullo schermo.

MESSAGGI D'ERRORE: UN MODO DI AGIRE DEL COMPUTER PER DIRE "NON CAPISCO"

Il comando PRINT dice semplicemente al computer di visualizzare sullo schermo qualsiasi cosa sia contenuta fra le virgolette. Al computer non importa quali lettere o simboli siano contenuti tra le virgolette; le parole non devono essere scritte correttamente o avere un senso. Provate le istruzioni qui sotto:

```
40 PRINT "AYE SAY HYE; U SAY BI."
RUN
```

Anche quando tra le virgolette c'è una frase contenente degli errori di ortografia o priva di senso il computer fa quello che gli è stato detto di fare.
Tuttavia, provate a scrivere male la parola PRINT come mostrato qui sotto e notate cosa succede:

```
50 PRIMT "I SAY HI; YOU SAY BYE."
```

Il computer vi da un Messaggio d'Errore. Il computer controlla solo quelle istruzioni, che sono fuori dalle virgolette perché queste sono esclusivamente usate dal computer. Le istruzioni contenute tra le virgolette sono per vostro uso, che il computer registra esattamente come gliele avete date. Spostate il cursore su di una riga vuota ma non cancellate lo schermo. Fate girare il programma e notate cosa succede.

Il Messaggio D'Errore 17 appare alla linea 50, la linea nella quale intenzionalmente avete scritto male la parola PRINT. Il Messaggio D'Errore 17 è chiamato "errore di sintassi". Questo vuol dire che le istruzioni non possono essere decifrate dal computer (per un elenco completo dei Messaggi d'Errore, vedere l'Appendice D).

Ci sono parecchi modi per correggere un Messaggio d'Errore. Il piú facile è quello di muovere il cursore sulla riga che contiene l'errore. Mettete il cursore sulla M di PRIMT cambiandola in N. Premete [Return] (in questo caso potete premere [Return] senza far caso alla posizione del cursore sulla riga, anche se si trova nel mezzo della parola PRINT). Nessun nuovo Messaggio d'Errore appare questa volta. Cancellate lo schermo e fate girare il programma. Lo schermo non dovrebbe mostrare nessun Messaggio d'Errore.

Un altro modo per correggere il Messaggio d'Errore è di cancellare la riga sbagliata. Per impratichirsi con questa tecnica, battete un'altra linea che contenga un errore intenzionale. Questa volta non mettete le virgolette nell'istruzione PRINT data qui sotto, poi battere RUN e LIST:

```
60 PRINT I ONCE HAD A PROGRAM CALLED BOZON
RUN
LIST
```

Un Messagio d'Errore apparirà quando premete [Return] e quando provate a girare e listare il programma. Per cancellare la riga errata, scrivete semplicemente il numero di linea e premete [Return]:

```
60
RUN
LIST
```

Adesso il programma gira e visualizza senza errori, anche se la linea 60 non contiene nessuna istruzione. La riga I ONCE HAD A PROGRAM CALLED BOZON è stata cancellata. Scrivendo il numero di linea e premendo [Return] si cancella completamente una riga dalla memoria del computer. Scrivete la linea correttamente come qui sotto:

```
60 PRINT "I ONCE HAD A PROGRAM CALLED BOZON"
RUN
```

PRINT: CREARE DELLE RIGHE VUOTE

Inserendo una riga vuota dopo la poesia la si renderà di piú facile lettura. Battete le seguenti istruzioni per creare una riga vuota tra le righe 30 e 40:

```
35 PRINT
RUN
LIST
```

Quando non c'è niente che segue il comando PRINT, il computer crea una riga vuota. Inserite una riga vuota tra le linee 50 e 60. Usate 55 come numero di riga e battete solo la parola PRINT dopo il numero.

?: ABBREVIAZIONE DI PRINT

Potete risparmiare tempo e fatica sostituendo il punto interrogativo (?) a PRINT. Provate la seguente linea di programma qui sotto:

```
70 ? "THAT RAN FROM DUSK UNTIL DAWN."
RUN
LIST
```

Il programma gira ugualmente con ? come con PRINT. Il punto interrogativo è un metodo piú conveniente e rapido. Per chiarezza, tutte le istruzioni PRINT di questo corso useranno la parola PRINT, ma è chiaro che la potete sostituire con il punto interrogativo.

LUNGHEZZA DI RIGA LOGICA

A volte tra le virgolette ci sono troppi caratteri per stare su una o due righe. Assicuratevi che il suono sia udibile sul vostro televisore o monitor prima di battere il seguente esempio:

```
80 PRINT "IT WOULDN'T RESPOND TO ESCAPE, BREAK, CONTROL,
OR LIST, AND IT WAS STILL RUNNING WHEN I TURNED OFF THE
SWITCH."
```

Quando il cursore raggiunge la terza riga il cicalino suona. Il cicalino vi avvisa che state raggiungendo la massima lunghezza di una riga di istruzione.

Un'istruzione non può essere piú lunga di tre righe di schermo. Questo limite è chiamato "riga logica" (adesso potete diminuire il volume se lo desiderate).

VISUALIZZAZIONE DELLO SCHERMO

Spesso le parole sono dimezzate nei posti sbagliati quando il cursore arriva alla fine della riga sullo schermo. Anche gli spazi tra le parole quando battete le linee di programma non sono gli stessi degli spazi tra le parole utilizzati dal computer quando fa girare il programma. Per evitare questi due tipi di problemi, decidete fin dall'inizio che formato deve avere ogni riga e battete separatamente una istruzione PRINT per ogni riga. Riscrivete la frase della linea 80 in formato poesia.

```
80 PRINT "IT WOULDN'T RESPOND"
90 PRINT "TO ESCAPE, BREAK, CONTROL, OR LIST,"
100 PRINT "AND IT WAS STILL RUNNING"
110 PRINT "WHEN I TURNED OFF THE SWITCH."
RUN
LIST
```

RIPRODURRE SIMBOLI GRAFICI

Potete usare i simboli grafici con l'istruzione PRINT per fare dei lavori artistici. Per inquadrare la poesia, battete le righe qui sotto. Usate [Control] [I] e [Control] [H] per creare le grafiche:

PRINT "↖": CANCELLARE LO SCHERMO

Potete mostrare il vostro programma in modo migliore assicurandovi che lo schermo sia pulito quando iniziate.
Battete un numero di righe, PRINT e le prime virgolette.
Premete leggermente il tasto [Esc] una sola volta, poi premete i tasti [Shift] [Clear] oppure i tasti [Control] [Clear]. Una freccia curva apparirà sullo schermi. Battete altre virgolette e premete [Return].
Fate girare e listate il programma:

```
5 PRINT "↖"
RUN
LIST
```

[CONTROL] [1] : FERMARE LO SCHERMO DI VISUALIZZAZIONE

Adesso il programma ha un'aspetto migliore, ma è troppo lungo per vederlo tutto in una volta sullo schermo. Quando il computer lista il programma, potete fermare le righe che si muovono verso l'alto e scompaiono dallo schermo premendo i tasti [Control] e [1]. Battete LIST. Usate due dita della mano sinistra per premere contemporaneamente i tasti [Control] e [1] ed un dito della mano destra per premere [Control]. I tasti [Control] e [1] iniziano ed arrestano la funzione LIST.

Creare un Loop Iterativo

GOTO, DIM, OUTPUT

I loop dicono al computer di tornare indietro e ripetere automaticamente le istruzioni del programma. Il comando GOTO vi risparmia la fatica di scrivere ripetutamente le medesime righe di istruzione. I comandi DIM e OUTPUT vi permettono di interagire con il vostro computer su una base di domande e risposte. Mettendo assieme questi tre comandi, potete dialogare con il vostro computer.

GOTO: LA MAPPA DEL COMPUTER

Il piú semplice ciclo d'iterazione (loop) è il GOTO. Il GOTO è sempre seguito da un numero che indica al computer a quale linea di programma andare per eseguire il prossimo comando. Avete bisogno di due comandi per creare un loop. Battete il programma qui sotto per produrre un loop infinito.

```
NEW
110 PRINT "CONGRATULATIONS!"
120 GOTO 110
RUN
```

Per arrestare questo loop infinito, spegnete il computer o premete il tasto [Break]. Quando arrestate il loop con il tasto [Break], appare uno di questi messaggi:

```
STOPPED AT LINE 110
```
oppure
```
STOPPED AT LINE 120
```

Il computer vi dice a che riga si trovava nell'istante in cui ha ricevuto il comando di arresto.

Virgola: Tabulare

Il loop GOTO esegue una grande mole di lavoro con solo due righe d'istruzione. Per rendere il programma piú interessante, listate il vostro programma, spostate il cursore nello spazio successivo a quello delle virgolette, inserite una virgola e premete [Return]. Fate girare il programma e notate l'effetto speciale:

```
LIST
110 PRINT "CONGRATULATIONS!",
120 GOTO 110
RUN
```

La virgola si comporta come una tabulazione. Ogni qualvolta il computer va alla linea seguente e visualizza CONGRATULATIONS!, si muove alla successiva posizione di tabulazione. Ricordatevi di arrestare il loop con il tasto [Break].

Punto e virgola: Unire

Il punto e virgola produce un altro effetto. Listate il programma, cambiate la virgola alla riga 110 con punto e virgola, premete [Return] e fate girare il programma:

```
LIST
110 PRINT "CONGRATULATIONS!";
120 GOTO 110
RUN
```

Il punto e virgola unisce il contenuto delle istruzioni PRINT senza lasciare spazi intermedi. Per avere degli spazi tra le parole, modificate la riga 110 che apparirà cosí:

```
110 PRINT "CONGRATULATIONS!  ";
RUN
```

Due punti: Separare

I due punti separano. Permettono a due istruzioni di essere scritte sulla medesima riga. Cambiate il punto e virgola della riga 110 con i due punti e aggiungete la frase seguente:

```
110 PRINT "CONGRATULATIONS!": PRINT "YOU JUST WON THE
LOTTERY."
RUN
```

Progredendo nella programmazione, diventa sempre piú importante risparmiare spazio nella memoria del computer. Consolidare diversi comandi su una linea con i due punti è un modo per risparmiare dei bytes inutilizzati nella memoria RAM. Per vedere quanta memoria avete risparmiato battete la frase seguente:

```
PRINT FRE (0)
```

Il computer risponderà con un numero. Riprogrammate la linea 110 in modo che le due istruzioni PRINT si trovino su due righe differenti.

```
110 PRINT "CONGRATULATIONS!"
115 PRINT "YOU JUST WON THE LOTTERY."
PRINT FRE (0)
```

Comparate i due numeri dei bytes ancora liberi. Il secondo numero è di due o tre unità meno del primo. Poiché la semplicità per un programmatore principiante è piú importante dei risparmi di memoria, le linee di programma di questa sezione avranno solo una istruzione per linea. La sola eccezione sarà l'istruzione PRINT, che inserisce una riga vuota tra segmenti del programma. Battete la linea indicata qui sotto per vedere l'effetto:

```
110 PRINT:PRINT "CONGRATULATIONS!"
RUN
```

DIM E INPUT: DIMENSIONARE ED INSERIRE VARIABILI STRINGA

Il computer deve essere programmato per rispondere ad una domanda. Potete usare un comando PRINT per fare una domanda ed un comando INPUT per mettere la risposta nel computer. Però, quando date al computer una risposta, il computer deve sapere dove collocarla; di solito in un posto chiamato "variabile" nella memoria RAM. Se la risposta è composta da lettere o da numeri o da tutti e due viene chiamata "variabile stringa". Il vostro computer ATARI 130XE deve sapere di quanto spazio ha bisogno per la risposta in modo da poterlo riservare. Questo processo viene chiamato dimensionare una "variabile stringa".

Il comando DIM (dimensionare) accompagna sempre il comando INPUT per le variabili stringa, perché DIM determina le dimensioni delle risposte. Per le variabili, le dimensioni si riferiscono al numero dei caratteri, spazi inclusi, che occorrono. Dovete indicare al computer il massimo numero di spazi che la risposta deve occupare.

Cambiate il programma del loop ad un programma che fa una domanda e attendete una risposta. Non c'è bisogno di riscrivere il programma; dovete soltanto aggiungere nuove righe—righe 10, 120, 130 e 140—come mostrato qui sotto. (Battendo la nuova riga 120, automaticamente si cancella la vecchia riga 120).

```
10 DIM ANSWER$ (100)
110 PRINT: PRINT "CONGRATULATIONS!"
115 PRINT "YOU JUST WON THE LOTTERY."
120 PRINT: PRINT "HOW DOES THAT MAKE YOU FEEL?"
130 INPUT ANSWER$
140 PRINT "I THOUGHT YOU WOULD SAY THAT."
RUN
```

La linea 10 indica al computer di riservare nella sua memoria uno spazio per una risposta lunga al massimo 100 caratteri. La variabile in questo programma è stata chiamata ANSWER. La variabile

memorizzerà lettere e numeri, perciò sarà una variabile stringa. Le variabili stringa sono designate con il simbolo del dollaro subito dopo l'ultima lettera del nome della variabile.

La linea 130 vi permette di inserire una risposta. Quando fate girare il programma, il computer visualizza la domanda sullo schermo e voi dovete battere la risposta. La risposta è memorizzata nella variabile stringa chiamata ANSWER$. Se l'istruzione DIM a linea 10 fosse stata omessa, un Messaggio d'Errore sarebbe apparso e l'istruzione di INPUT non avrebbe funzionato.

?: Cortesia dell'INPUT

Fate girare di nuovo il programma. Due punti interrogativi appariranno sullo schermo. Il secondo punto interrogativo si troverà sulla linea vicino al margine sinistro. Listate il programma e notate che avete battuto solo un punto interrogativo. Il comando INPUT mette sempre un punto interrogativo sullo schermo. Provate la seguente variazione alla linea 120:

```
120 PRINT "HOW DOES THAT MAKE YOU FEEL";
```

Fate girare il programma e rispondete quando il computer vi pone la domanda. Adesso c'è solo un punto interrogativo e la vostra risposta segue immediatamente la domanda sulla medesima riga. Create altri dialoghi dimensionando altre variabili stringa e inserendo altre istruzioni INPUT. L'istruzione DIM deve trovarsi all'inizio del programma:

```
20 DIM DATE$ (25)
140 PRINT:PRINT "WHEN WOULD YOU LIKE TO COME AND PICK UP
YOUR PRIZE";
150 INPUT DATE$
RUN
```

Variabili Stringa nell'Istruzione PRINT

Il programma del computer ora vi pone due domande, ma non risponde all'ultima domanda. Per avere una risposta, inserite la variabile stringa nell'istruzione PRINT nel seguente modo:

```
160 PRINT "I'M SORRY, BUT OUR OFFICES ARE ALWAYS CLOSED
ON ";DATE$;". TOO BAD!"
```

Il punto e virgola unisce la variabile stringa delle due frasi tra virgolette. Fate girare il programma. Se le parole, non sono separate correttamente, comparate la vostra linea con quella sopra.

Probabilmente vi siete dimenticati uno spazio dopo la L di IL oppure il punto e lo spazio prima di TOO BAD!. Questi spazi sono molto importanti. Fate pratica con un'altra variabile stringa:

```
30 DIM NAME$ (1)
170 PRINT "BY THE WAY, WHAT IS YOUR NAME";
180 INPUT NAME$
190 PRINT "WELL, "; NAME$;", I BET YOU WOULD LIKE TO KNOW
HOW MUCH YOU WON. FIRST YOU HAVE TO ANSWER A QUESTION."
```

Fate girare il programma. Sebbene abbiate scritto il nome completo, il computer visualizzerà solo la vostra iniziale. Questo avviene perché l'area dimensionata nella memoria RAM per il nome era troppo piccola. Il nome della maggior parte delle persone è piú lungo di un carattere. Cambiate la linea 30 con un numero di spazi maggiori e fate girare il programma:

```
30 DIM NAME$ (25)
RUN
```

Ingresso delle Variabili Numeriche

Fino adesso avete lavorato con variabili stringa alfanumeriche composte da lettere, numeri o tutti e due. Per esempio, il computer accetterà il nome R2-D2 o 007 come una variabile stringa. Tuttavia il numero come nome sarà riconosciuto solo come tale e non come numero di un problema matematico. Adesso provate delle variabili numeriche che possono essere usate nei calcoli matematici. Le variabili numeriche non hanno bisogno del comando DIM o simbolo del dollaro. Scrivete le seguenti linee di programma:

```
200 PRINT:PRINT "HOW OLD ARE YOU";
210 INPUT AGE
220 PRIZE=AGE*1000
230 PRINT:PRINT "YOU HAVE JUST WON $"; PRIZE;" FROM THE
LOTTERY. YOU CAN COLLECT DURING OFFICE HOURS."
```

In questo programma, l'età che avete dato è memorizzata nella variabile numerica chiamata AGE. La linea 220 crea un'altra variabile chiamata PRIZE e permette al calcolatore incorporato nel computer di calcolare la vincita che è $1000 moltiplicato per l'età del vincitore (per il computer significa moltiplicare). Il programma esegue il calcolo matematico e memorizza la risposta nella variabile PRIZE. La linea 230, che inserisce il valore numerico nell'istruzione PRINT come una variabile stringa, vi dà la risposta giusta.

I LOOPS DI INPUT

Per ripetere il dialogo con il computer, aggiungete ancora un comando di loop al programma. Un'istruzione GOTO alla fine farà ripetere al computer il programma dall'inizio. Per facilitare la lettura del programma, usate un'istruzione REM per indicare l'inizio del dialogo nel programma. Un'istruzione REM (remark) funziona come talloncino per il programmatore. Il computer non esegue un comando REM, lo stampa soltanto quando listate il programma.

```
100 REM *** CONVERSATION LOOP ***
240 GOTO 100
```

Il computer deve ritornare alla linea 100 piuttosto che alla linea 10, perché non può ritornare all'istruzione DIM per le variabili stringa. Se fa il loop sulle medesime istruzioni DIM, riceverete un Messaggio d'Errore.

L'Uso di Numeri Casuali e Funzioni Matematiche

RND, +,-,*,/

Inizialmente i computers erano costruiti per fare i calcoli rapidamente e facilmente. Per sfruttare le capacità del computer e trovare delle risposte matematiche in pochi millisecondi, dovete sapere come comunicare con il computer.

NUMERI

Battete l'istruzione qui sotto e premete [Return] :

```
PRINT 10
```

Il computer deve visualizzare il numero 10. Assicuratevi che usate i numeri 1 e 0 e non le lettere. Fate pratica a stampare i seguenti numeri:

```
PRINT 1000000000
PRINT -100000000
```

Usate il segno meno (-) sul tasto della freccia verso l'Alto [^ -] per indicare i numeri negativi. Non usate virgole con i numeri. Battete le istruzioni qui sotto e notate cosa succede quando usate le virgole:

```
PRINT 9,876,543,210
PRINT 9, 876, 543, 210
```

In ambedue i casi il computer interpreta le virgole come separatori di una serie di numeri. Separa i numeri sullo schermo secondo l'attuale predisposizione di tabulazione. Per il computer, il 9 non è 9 miliardi, è solo il numero 9 seguito da una serie di altri numeri.

NOTAZIONE SCIENTIFICA

Il computer non capisce le virgole ma capisce gli esponenti. Spesso trasforma automaticamente un numero grande in forma esponenziale. Provate i seguenti numeri:

```
PRINT 99999999999
PRINT 55555555555
PRINT 11111111111
PRINT -11111111111
PRINT -98765432112
```

Questi numeri sono grandi o abbastanza piccoli per il computer, che preferisce riscriverli nella notazione scientifica. Non è necessario avere familiarità con la notazione scientifica per capire il computer o questo capitolo.

La notazione scientifica esprime un numero grande, tra 0 e 10, moltiplicato per una potenza di 10. L'esponente specifica la potenza di 10. Nel seguente esempio, E + 13 significa che l'esponente è 13:

2.5E + 13 = 2.5 x 10^{13} = 25000000000000

Potete usare gli esponenti per comunicare con il vostro computer. L'accento circonflesso sul tasto della Freccia a destra [^] è il simbolo degli esponenti. Dovete usare il tasto [Shift] per stampare l'accento circonflesso. Provate i seguenti calcoli:

```
PRINT 2 ^ 1
PRINT 2 ^ 2
PRINT 2 ^ 3
PRINT 2 ^ 4
PRINT 2 ^ 64
```

La prima notazione è 2 alla prima potenza, la seconda 2 alla seconda potenza e cosí via. L'ultima notazione è 2 alla sessantaquattresima potenza, un numero abbastanza grande per il computer, che lo esprime in notazione scientifica.

A meno che siate un fisico, che misura il tempo degli elettroni nella loro orbita od un astronomo, che calcola le dimensioni dell'universo, raramente avrete bisogno della notazione scientifica. Ma se ne avete bisogno, il computer sarà capace di eseguire calcoli anche con questi numeri giganteschi.

IL COMPUTER COME CALCOLATORE

Il computer può eseguire le stesse funzioni come calcolatore. Usate il segno piú (+) sul tasto della Freccia verso sinistra [+] per battere l'istruzione qui sotto:

```
PRINT 1+1
```

Quando premete [Return], il computer immediatamente vi dà la risposta, come un calcolatore. Fate degli esercizi di addizione. Usate numeri piccoli o grandi e provate a sommare una lunga serie di numeri. Fate delle prove con molte varianti.

Usate il segno (-) sul tasto della Freccia verso l'alto [-] per sottrarre i numeri. Provate le tre versioni dei problemi qui sotto:

```
PRINT 4 - 1
PRINT 4-1
PRINT4-1
```

Alternate la stessa risposta ogni qualvolta premete [Return]. La spaziatura nei problemi di matematica non è importante per il computer. Provate con dei problemi vostri. Fate dei problemi lunghi con addizioni e sottrazioni.

Il segno di moltiplicazione, l'asterisco (*), si trova sul tasto della Freccia a Destra [:*]. Il segno della divisione è la barra (/) sul tasto del punto interrogativo [?/]. Battete le seguenti istruzioni:

```
PRINT 2 * 2
PRINT (2*2)
PRINT 6 / 3
PRINT (6/3)
```

Il computer non solo capisce l'uso delle parentesi in matematica, ma le usa quando i problemi diventano piú complessi. Notate che cosa succede in questo problema con e senza parentesi:

```
PRINT 3* (2+2)
PRINT 3*2+2
```

La risposta al primo problema è 12, al secondo è 8. Nel primo problema il computer esegue 2 piú 2 e poi moltiplica per 3 per ottenere 12. Nel secondo, il computer moltiplica 3 per 2 prima e poi somma 2 per ottenere 8. Ogni qualvolta il computer trova una parentesi in un problema di matematica, esegue prima i calcoli dentro le parentesi e poi finisce il resto dei calcoli.

Provate i problemi qui sotto per capire alcuni aspetti interessanti di come funziona il computer. Provate a indovinare la risposta prima di premere [Return] .

```
PRINT (2+2)*3
PRINT 2+2*3
```

Nel primo problema, il computer esegue prima il calcolo dentro le parentesi. Nel secondo, esegue prima la moltiplicazione e poi l'addizione. Il computer esegue queste funzioni matematiche secondo le regole di ordine: primo calcolo dentro le parentesi, secondo le funzioni esponenziali, terzo, le funzioni di moltiplicazione e divisione come appaiono nel problema da sinistra a destra ed infine le somme e le sottrazioni da sinistra a destra. Le regole sono riassunte nelle seguenti tavole:

Ordine Matematico di Esecuzione

1.	( )	Calcolo tra parentesi	
2.	^	Funzioni esponenziali	
3.	*	Moltiplicazioni	In ordine da comparizione
	/	Divisioni	Da sinistra a destra
4.	+	Somme	In ordine di comparizione
	-	Sottrazioni	Da sinistra a destra

NUMERI CASUALI

Il computer può eseguire altre funzioni che il vostro calcolatore presumibilmente non può fare. Per esempio, il computer può generare numeri casuali. Battete il programma qui sotto:

```
NEW
10 PRINT RND (0)
20 GOTO 10
RUN
```

RND è il comando per generare numeri casuali che il loop infinito nel programma genera fino a che non lo fermate. Ricordatevi di terminare il loop con il tasto [Break]. Per apportare cambi al programma, potete listarlo e usare i tasti del cursore per inserire dei caratteri, piuttosto che riscrivere delle linee intere. Provate i programmi qui di seguito:

```
10 PRINT RND (1)
RUN
10 PRINT RND (123)
RUN
10 PRINT RND (50)
RUN
10 PRINT RND (50000)
RUN
```

Tutte e quattro le varianti della linea 10 generano dei numeri casuali tra 0 e 1. Il punto decimale viene sempre prima della prima cifra in numero casuale. I pochi numeri casuali che hanno un numero alla sinistra del punto, sono pure compresi tra 0 e 1 ma sono cosí piccoli che il computer li esprime nella notazione scientifica.

I numeri tra parentesi sono chiamati "variabili fittizie". Non importa che numero sia usato come variabile fittizia, ma è importante che ci siano le parentesi e che racchiudano qualcosa (sia numero o lettera). Per facilità di battitura, lo 0 è abitualmente messo nella posizione di variabile fittizia. Cambiate la linea 10 come mostrato qui sotto:

```
10 PRINT (RND(0) * 10)
RUN
10 PRINT (RND(0) * 100)
RUN
10 PRINT (RND(0) * 1000)
RUN
```

Ogni programma genera una gamma di differenti numeri casuali. PRINT (RND(0)* 10) genera numeri fino a 10 perché il computer viene istruito a moltiplicare il numero casuale per 10. Moltiplicando per dieci, sposta il punto decimale di un posto. PRINT (RND(0)* 100) moltiplica per 100 e sposta il punto decimale di due posti. PRINT (RND(0)* 1000) moltiplica per 1000 e sposta il punto decimale di tre posti. Se volete, potete moltiplicare con numeri piú grandi per generare dei numeri casuali piú grandi.

Poiché i numeri lunghi con tante cifre dopo il punto decimale sono ingombranti, il computer ha un'istruzione che gli dice di visualizzare solo gli interi, che sono numeri senza punti decimali. L'istruzione INT indica al computer di ignorare qualsiasi numero dopo il punto decimale. Riprogrammate le tre varianti della linea 10 e comparate i risultati:

```
10 PRINT INT(RND(0)*10)
RUN
10 PRINT INT(RND(0)*100)
RUN
10 PRINT INT(RND(0)*1000)
RUN
```

Il programma genera numeri nella medesima gamma di prima, ma sono piú facili da leggere senza cifre dopo il punto decimale.

Per generare i numeri in una gamma piú specifica provate gli esempi qui sotto:

```
10 PRINT INT (RND(0)*3)
RUN
10 PRINT INT (RND(0)*12)
RUN
10 PRINT INT (RND(0)*25)
RUN
```

Il programma genera numeri casuali, che sono sempre inferiori di un numero rispetto a quelli per cui vengono moltiplicati. La prima linea 10 genera i numeri 0,1 e 2. Per generare numeri casuali 0,1, 2 e 3 il programma dovrebbe essere scritto cosí:

```
10 PRINT INT (RND(0)*4)
RUN
```

Per generare solo i numeri 1, 2 e 3 il programma dovrebbe essere cosí:

```
10 PRINT INT (RND(0)*3)+1
RUN
```

Per generare tre numeri incominciando da 20, scrivete il programma cosí:

```
10 PRINT INT (RND(0)*3)+20
RUN
```

Gioco con i Numeri Casuali

I programmi con i numeri casuali sono molto flessibili. Li potete perfino usare per fare dei giochi con il computer. Battete il seguente programma. Ricordatevi che per ottenere la freccia piegata alla linea 5 dovete premere [Esc], tenere premuto [Shift] o [Control] e premere [Clear]:

```
NEW
1 REM *** NUMBER.GAM ***
5 PRINT "↖"
10 SECRETNUM=INT(RND(0)*3)+1
20 PRINT: PRINT "I AM THINKING OF A NUMBER, EITHER 1, 2,
OR 3. TRY TO GUESS IT."
30 INPUT GUESS
40 IF GUESS=SECRETNUM THEN PRINT "YOU WON."
50 IF GUESS<>SECRETNUM THEN PRINT "YOU LOST."
60 GOTO 10
```

La linea 10 assegna il numero casuale alla variabile numerica SECRETNUM. La linea 30 invita l'utente a rispondere all'indovinello e assegna questo numero alla variabile numerica GUESS (ricordatevi che le variabili numeriche non devono essere dimensionate o etichettate alla fine come le variabili stringa). La linea 40 compara la risposta con il numero segreto. Se sono uguali il computer visualizza "YOU WON". Anche la linea 50 compara la risposta con il numero segreto. Se non sono uguali (il simbolo< >significa non uguale), il computer visualizza "YOU LOST". La linea 60 chiude il ciclo iterativo (loop) per incominciare il gioco di nuovo (il prossimo capitolo spiega l'espressione IF-THEN in dettaglio).

PROBLEMI DI MATEMATICA

Le funzioni matematiche del computer possono essere usate sia per lavoro che per gioco. Se siete un cuoco e preparate dei banchetti, potreste usare il computer per arricchiare le vostre ricette. Per esempio, supponiamo che cercate di sapere quanti etti di scaloppe di mare dovete acquistare per servire le Coquilles St. Jacques ad un pranzo per 62 ospiti. La ricetta vi indica che vi occorre 1.5 libbra per 5 persone. Il programma qui sotto vi dice quante libbre acquistare:

```
NEW
1 REM *** COQUILLE ***
10 PRINT "♥"
20 GUESTS=62
30 POUNDSTOBUY= 1.5/5 * GUESTS
40 PRINT:PRINT "BUY ";POUNDSTOBUY;" POUNDS OF
SCALLOPS."
50 END
```

Il programma dà la risposta (18.6 libbre di scaloppe), ma il calcolatore avrebbe dato lo stesso risultato con meno lavoro. Per rendere il programma piú utile, cambiate il numero degli ospiti inserendo un'espressione INPUT. Battete le linee aggiuntive qui sotto:

```
15 PRINT:PRINT "HOW MANY GUESTS DO YOU EXPECT?"
20 INPUT GUESTS
```

Fate girare il programma diverse volte, con differenti numeri di ospiti. La quantità di scaloppe cambierà ogni volta. Per 200 ospiti occorrono 60 libbre di scaloppe; per 436 ospiti, 130.8 libbre. La funzione INPUT rende il programma piú pratico.

Prendere Decisioni e Risolvere Problemi

IF-THEN, FOR-NEXT

I comandi IF-THEN e FOR-NEXT vi permettono di scrivere programmi che imitano l'approccio usato dall'uomo per prendere decisioni e risolvere problemi. Questi comandi sono particolarmente utili per giochi e puzzle logici e vi permettono di fare delle scelte al posto del computer.

COMANDI IF-THEN

Per fare pratica con l'espressione IF-THEN, battete il programma seguente:

```
NEW
1 REM *** BRNPROBE.QZ ***
5 PRINT "↖"
10 DIM RAIN$(3)
20 PRINT:PRINT "YES OR NO, IF IT WERE RAINING OUTSIDE,
WOULD YOU GO OUT WITH AN UMBRELLA";
30 INPUT RAIN$
40 IF RAIN$="YES" THEN PRINT "YOU HAVE A FORMIDABLE IQ."
50 IF RAIN$="NO" THEN PRINT "YOU ARE A BORN RISK TAKER."
```

Il Quiz BRAINPROBE valuta la vostra risposta. Nella linea 40, se la risposta memorizzata dalla variabile stringa RAIN$ è "SI"/YES, il computer visualizza il messaggio dell'IQ. Se la risposta non è SI, il computer legge la linea seguente, la 50, e valuta di nuovo il contenuto della stringa variabile, RAIN$. Se la risposta è "NO", il computer visualizza il secondo messaggio. Se la risposta non è né SI né NO, il programma si ferma, perché non contiene nessuna istruzione per obbedire ad una risposta indefinita. Provateci.

Un modo per incoraggiare una risposta attesa è di creare un loop. Inserite nel programma la linea qui sotto:

Valutazione con l'IF-THEN

Un altro modo per stimolare una risposta corretta è di fornire dei suggerimenti. Il seguente programma usa delle variabili numeriche per ottenere la risposta giusta.

```
NEW
1 REM *** NUMBER.QZ ***
5 PRINT "↖"
10 SECRETNUM=INT(RND(0)*10)+1
20 PRINT:PRINT "GUESS A SECRET NUMBER BETWEEN 1 AND 10."
```

```
30 PRINT
40 PRINT "YOUR GUESS";
50 INPUT GUESS
60 PRINT
70 IF GUESS=SECRETNUM THEN PRINT "YOU GOT IT!":END
80 IF GUESS<SECRETNUM THEN PRINT "TOO LOW. TRY AGAIN.":
GOTO 40
90 IF GUESS>SECRETNUM THEN PRINT "TOO HIGH.TRY
AGAIN.":GOTO 40
```

Le line 80 e 90 valutano se la risposta è minore o maggiore del numero segreto. L'espressione PRINT indica se la prossima risposta dovrà essere piú grande o piú piccola. I comandi GOTO alle linee 80 e 90 creano un'iterazione infinita se date sempre la risposta sbagliata.

Terminare il Programma

Il Quiz NUMBER (numero) è programmato per arrestarsi solo dopo che avete scoperto il numero segreto. Quando date la risposta giusta, la linea 70 dà al computer l'istruzione di terminare. L'END ferma il programma e la parola READY appare sullo schermo.

Intrappolare gli errori

Se accidentalmente per risposta battete una lettera invece di un numero, il computer vi dà un Messaggio d'Errore e si arresta di colpo. Create un errore intenzionale battendo una lettera o premendo solo [Return]. Per evitare di arrestare il programma potete usare il comando TRAP per intrappolare il Messaggio d'Errore. Aggiungete le seguenti linee e fate girare il programma di nuovo:

```
45 TRAP 100
100 PRINT:PRINT "PLEASE ENTER A NUMBER ONLY."
110 GOTO 30
```

Alla linea 45 il comando TRAP indica al computer di non fermarsi quando c'è stato un errore e lo manda alla line 100. La linea 100 indica al computer di visualizzare le istruzioni per correggere l'errore. L'istruzione TRAP viene sempre prima dell'istruzione INPUT e contiene sempre il numero della linea che risolve il problema.

Scrivere dei Quiz con IF-THEN

Il programma può facilmente suggerire la risposta corretta quando questa è un numero o una data. Il seguente programma usa l'istruzione IF-THEN ed il comando TRAP per suggerire la risposta.

```
NEW
1 REM *** LOVELACE.QZ ***
5 PRINT "↖"
10 PRINT:PRINT "ADA LOVELACE, DAUGHTER OF THE POET LORD
BYRON, WAS MATHEMATICALLY BRILLIANT."
20 PRINT
30 PRINT "IN WHAT YEAR DID SHE WRITE HER AMAZINGLY
ACCURATE DESCRIPTION OF THE FUTURE USES OF THE
COMPUTER";
40 TRAP 200
50 INPUT GUESS
60 IF GUESS=1842 THEN GOTO 100
70 IF GUESS<1842 THEN GOTO 110
80 IF GUESS>1842 THEN GOTO 120
100 PRINT: PRINT "CONGRATULATIONS! YOU GUESSED THE YEAR
CORRECTLY.":END
110 PRINT:PRINT "THAT WAS TOO EARLY. TRY AGAIN.":GOTO 20
120 PRINT:PRINT "THAT WAS TOO LATE. TRY AGAIN.":GOTO 20
200 PRINT:PRINT "PLEASE ENTER A NUMBER ONLY."
210 GOTO 20
```

Nel Quiz Lovelace la posizione delle espressioni PRINT associate alle istruzioni IF-THEN è differente da quella dei Quiz Provatesta e Numeri. Questa differenza dimostra che ci sono diversi modi per raggiungere gli stessi risultati.

Computer Bugs

L'istruzione TRAP ha reso il Quiz Lovelace a prova di errore, ma non è ancora perfetto. Questo perché il computer valuta la data come numero: infatti accetterà 1842.78 come errato ma 1842,78 come corretto. La maggior parte dei programmi hanno dei "bugs", ossia problemi e carenze. Quando avrete imparato a scoprire i bugs e a correggerli avrete imparato a programmare. Ogni principiante troverà molti bugs e farà molti errori. Per diventare un buon programmatore, studiate questo manuale, consultate la Guida Riferimento del BASIC ATARI ed anche un programmatore piú esperto di voi. Imparerete ad identificare i bugs ed evitarli nei programmi che farete in futuro.

LOOP FOR-NEXT: IL LOOP CONTATORE

Avete già una certa familiarità con il loop infinito GOTO. Un altro tipo di loop è il FOR-NEXT. Il loop FOR-NEXT è un loop contatore che non è infinito. Battete NEW e il seguente programma:

```
NEW
10 FOR X=1 TO 4
20 PRINT "POTATO"
30 NEXT X
RUN
```

POTATO viene visualizzato sullo schermo quattro volte. Cambiate la linea 10 in questo modo:

```
10 FOR X=1 TO 7
```

Questa volta il programma visualizza POTATO sette volte. Il computer gira sette volte sulle linee 10, 20 e 30. FOR indica al computer quante volte deve girare, e NEXT indica al computer di ritornare al principio e ricominciare. NEXT è simile a GOTO. La X è una variabile. Potete usare qualsiasi cosa per rappresentare la variabile. Provate questo nome come variabile:

```
10 FOR NUM=1 TO 7
30 NEXT NUM
```

Quando fate girare il programma non c'è nessuna differenza con il precedente programma. Cambiate ancora il nome della variabile:

```
10 FOR JKL=1 TO 7
30 NEXT JKL
```

JKL è una sigla senza significato usata come variabile numerica nel loop FOR-NEXT. Fate girare il programma per rendervi conto che gira come prima. Adesso aggiungete questa linea:

```
15 PRINT JKL,
RUN
```

L'espressione PRINT alla linea 15 mostra il valore della variabile (la virgola rende piú facile la lettura). Ogni qualvolta il computer ripete il loop FOR-NEXT, la variabile assume il valore del numero successivo della serie come specificato a linea 10. Per la prima volta la variabile è 1; la seconda è 2, e cosí via. L'ultimo numero nell'espressione FOR controlla il numero di volte che il computer gira nel programma. Cambiate il numero della linea 10 come qui sotto:

```
10 FOR JKL=1 TO 50
RUN
10 FOR JKL=1 TO 200
RUN
10 FOR JKL=1 TO 500
RUN
```

Punto di Partenza

Listate il programma. Il primo numero nella linea FOR è il punto di partenza del conteggio e l'ultimo numero è il punto di arresto. Anche i numeri negativi possono essere usati come punti di partenza. Provate queste modifiche alla linea 10.

```
10 FOR JKL=1 TO 5
RUN
10 FOR JKL=0 TO 5
RUN
10 FOR JKL=3 TO 5
RUN
10 FOR JKL=-10 TO 5
RUN
```

STEP: Conteggio Incrementale

Listate il programma, eliminate l'espressione PRINT alla linea 20 e la virgola alla linea 15 e fate girare il programma. Il computer conta e visualizza i numeri molto rapidamente. Usate il comando STEP per fare contare il computer in modo incrementale. Provate questo:

```
10 FOR JKL=0 TO 500 STEP 5
RUN
10 FOR JKL=0 TO 500 STEP 2
RUN
10 FOR JKL=0 TO 500 STEP 100
RUN
10 FOR JKL=0 TO 500 STEP 7
RUN
```

Il computer conterà con qualsiasi sequenza gli specifichiate.

Conteggio all'Indietro

Il computer può contare all'indietro se usate il comando STEP-1 e la giusta sequenza di numeri (dai piú grandi ai piú piccoli) per iniziare ed arrestare il conteggio. Per esempio:

```
10 FOR JKL=500 TO 0 STEP -1
RUN
10 FOR JKL=10 TO 0 STEP -1
RUN
10 FOR JKL=-1 TO -19 STEP -1
RUN
```

Il computer può anche contare all'indietro in incrementi:

```
10 FOR JKL=500 TO 0 STEP -20
RUN
10 FOR JKL=500 TO 0 STEP -3
RUN
10 FOR JKL=0 TO -500 STEP -50
RUN
```

Potete anche istruire il computer di iniziare ed arrestarsi con qualsiasi numero specifichiate:

```
10 FOR JKL=500 TO 300 STEP -10
RUN
10 FOR JKL=25 TO 0 STEP -1
RUN
```

Adesso sapete come istruire il computer a contare avanti ed indietro, a contare consecutivamente ed incrementalmente, ad iniziare e ad arrestarsi con numeri specifici.

Il Loop "Sandwich" FOR-NEXT

Listate il programma. FOR si trova nella linea in alto del programma e NEXT nella linea in basso. Qualsiasi cosa vogliate far fare al computer questo si trova in mezzo al programma come in un sandwich. Battete la linea qui sotto:

```
10 FOR JKL=1 TO 5
20 PRINT "AVOCADO"
```

Il computer eseguirà qualsiasi istruzione o diverse istruzioni comprese tra FOR e NEXT tante volte quante ne avete specificate. Provate questo programma:

```
16 PRINT "CHEESE"
17 PRINT "MAYONNAISE"
18 PRINT "MUSTARD"
19 PRINT "TOMATO"
21 PRINT "BACON BITS"
22 PRINT "LETTUCE"
23 PRINT:PRINT
RUN
```

Il computer visualizzerà e conterà troppo rapidamente per leggere lo schermo. Nonostante tutto, visualizzerà le espressioni PRINT cinque volte come istruito dall'espressione FOR-NEXT. Altre istruzioni, come calcoli matematici ed espressioni di INPUT, possono essere parte del loop a sandwich FOR-NEXT.

Loop di Ritardo

Cancellate tutte le espressioni PRINT affinché non ci sia nulla nel mezzo del loop FOR-NEXT, ad eccezione di FOR e NEXT. Fate girare il programma e notate che cosa succede:

```
15
16
17
18
19
20
21
22
23
LIST
RUN
```

Non succede niente. Cambiate il numero a linea 10 e fate molta attenzione.

```
10 FOR JKL=1 TO 500
RUN
```

La parola Ready tarda alcuni secondi prima di apparire. Cambiate di nuovo la linea 10.

```
10 FOR JKL=1 TO 5000
RUN
```

Questa volta la parola di sollecito READY impiega molto piú tempo prima di apparire. Il computer conta ma non visualizza i calcoli. Il processo è simile a quando contate a mente. Il tempo che il sollecito READY impiega per apparire sullo schermo è il tempo che il computer impiega per contare fino a 5000.

Le iterazioni (loop) FOR-NEXT sono degli strumenti eccellenti per impedire al computer di proseguire. In realtà le iterazioni FOR-NEXT, che sono usate frequentemente per questo scopo, sono chiamate "delay loops" e il nome della variabile comunemente usato è DELAY. Riscrivete l'iterazione FOR-NEXT usando DELAY come nome della variabile e numeri diversi per l'istruzione FOR.

```
NEW
10 FOR DELAY=1 TO 300
20 NEXT DELAY
LIST
RUN
```

A volte il loop di ritardo viene inserito sulla medesima linea di programma:

```
NEW
10 FOR DELAY=1 TO 300:NEXT DELAY
LIST
RUN
```

Programmi Dimostrativi

I programmi qui sotto usano le iterazioni FOR-NEXT in diversi modi. Il primo programma usa l'iterazione FOR-NEXT come un semplice loop di ritardo per visualizzare sullo schermo la parola HI solo per il tempo di leggerla prima che la linea 30 cancelli lo schermo.

```
NEW
1 REM *** DLAYLOOP ***
5 PRINT " ↖ "
10 PRINT "HI"
20 FOR DELAY=1 TO 800:NEXT DELAY
30 PRINT " ↖ "
40 PRINT "BYE"
50 FOR DELAY=1 TO 800:NEXT DELAY
```

Il seguente programma usa una variabile numerica nel loop FOR-NEXT. Usa pure un comando TRAP che fa ritornare il computer alla linea precedente senza dare nessun messaggio specifico d'errore:

```
NEW
1 REM *** HOWHIGH? ***
10 DIM A$ (1), HH$(1)
20 PRINT "↖"
30 PRINT:PRINT "HOW HIGH DO YOU WANT TO COUNT";
40 TRAP 30
50 INPUT HH
55 HH$=STR$(HH): IF HH$="0" THEN GOTO 30
60 FOR COUNT=1 TO HH
70 PRINT COUNT
80 NEXT COUNT
90 PRINT :PRINT "PLEASE ANSWER (Y/N). WOULD YOU LIKE TO
COUNT AGAIN";
100 TRAP 90
110 INPUT A$
120 IF A$="Y" THEN GOTO 30
130 IF A$="N" THEN PRINT:PRINT "BYE":END
140 GOTO 90
```

L'ultimo programma parafrasa una vecchia canzone di rock and roll e usa la nidificazione delle iterazioni FOR-NEXT. Un'iterazione nidificata FOR-NEXT è una piccola iterazione di ritardo dentro un'iterazione FOR-NEXT piú grande. Il programma usa OR per creare delle condizioni multiple nell'espressione IF-THEN:

```
NEW
1 REM *** CLOCKRCK ***
5 PRINT "↖"
10 FOR X=1 TO 9
20 PRINT X;
30 PRINT " O'CLOCK"
40 FOR DELAY=1 TO 500:NEXT DELAY
50 IF X=3 OR X=6 OR X=9 THEN PRINT "ROCK!": FOR PAUSE=1
TO 500:NEXT PAUSE
60 NEXT X
70 PRINT:PRINT "WE'RE GOING TO ROCK"
80 PRINT "AROUND THE CLOCK"
90 PRINT "TONIGHT!"
```

Produrre Suoni e Grafiche

SOUND, SETCOLOR, COLOR

Creare dei suoni e grafici su alcuni computer è molto complicato, ma non sull'ATARI. Il comando SOUND del BASIC ATARI, combinato con alcune semplici tecniche di programmazione è tutto quello che vi occorre. I suoni e la grafica aggiungono una nuova dimensione ai vasti programmi BASIC che consentono di eseguire giochi, detonazioni, schiocchi, melodie musicali, canzoni e variopinte immagini grafiche.

PER SUONARE

Il vostro ATARI 130XE può produrre fino a 4 suoni contemporaneamente. I quattro registri sonori e le voci sono numerati 0, 1, 2 e 3. Per selezionare la prima voce, battere SOUND 0; per la seconda SOUND 1; per la terza, SOUND 2; e per la quarta, SOUND 3.

Il comando SOUND nel BASIC ATARI controlla quattro elementi:

voce (0-3)
tonalità (0-255)
distorsione (0-14)
volume (0-15)

La tonalità, o frequenza, del suono è determinata da un numero compreso tra 0 e 255, per un totale di 256 frequenze tra le quali potete scegliere. Il valore di tonalità è il secondo numero dell'istruzione SOUND. Per esempio, SOUND 1, 50 specifica la seconda voce con una tonalità di 50. Assicuratevi che il volume del vostro televisore o monitor sia regolato bene, poi battete:

```
SOUND 1,50,0,8
```

Premete [Return]. Una grande esplosione, vero? Per arrestare il suono, diminuite il volume del televisore oppure battete le istruzioni qui sotto e premete [Return]:

```
END
SOUND 1,0,0,0
```

La purezza (o distorsione) del rumore è determinata da un numero pari tra 0 e 14. Nell'istruzione SOUND la purezza del suono è rappresentata dal terzo numero. Provate questo:

```
SOUND 1,50,10,8
```

Il numero 10 produce un tono puro senza distorsioni. Per immettere un po' di distorsione, cambiate il 10 in 06.

```
SOUND 1,50,06,8
```

Il computer sembra che voglia decollare. Battete END prima che i vostri vicini incomincino a lamentarsi.

L'ultimo numero nell'istruzione SOUND controlla il volume. Il numero deve essere compreso tra 0 e 15. Il numero 8 va bene nella maggior parte dei casi. Se andate oltre il numero 12 rischiate di danneggiare l'altoparlante della TV e il vostro udito.

Per provare un'armonia a quattro suoni battete:

```
SOUND 0,50,10,8
SOUND 1,100,10,8
SOUND 2,150,10,8
SOUND 3,200,10,8
```

Battete END per terminare il coro.

Suonare con le Variabili

Le variabili con l'istruzione SOUND aggiungono flessibilità ai vostri programmi. Usando le variabili, potete programmare il computer a cambiare voce, tonalità, distorsioni e volume di suoni sostenuti. Battete e fate girare questo programma:

```
NEW
10 REM * SET VARIABLES FOR SOUND VALUES
20 VOICE=0:PITCH=100:TONE=8:VOL=8
30 SOUND VOICE,PITCH,TONE,VOL
40 GOTO 20
RUN
```

Per arrestare il suono, premete il tasto [Break] e battete END. Per sostenere un suono, dovete ripetere l'istruzione SOUND nel programma. I due metodi piú usati sono il loop FOR-NEXT o il loop GOTO come quello nell'esempio qui sopra. Il seguente programma usa una variabile per la tonalità in un loop FOR-NEXT per produrre l'intera gamma delle tonalità disponibili nel computer:

```
NEW
10 REM * SOUND EFFECTS WITH FOR-NEXT LOOP
20 VOICE=0:PITCH=0:TONE=10:VOL=8
30 FOR PITCH=0 TO 255
40 SOUND VOICE,PITCH,TONE,VOL
50 NEXT PITCH
RUN
```

Variando il volume in un programma, si produrrà una diversità di suoni. Portate il VOL=8 al VOL=0 e premete Return . Poi aggiungete la seguente linea:

```
35 VOL=INT(RND(0)*16)
```

Questa linea seleziona casualmente un valore tra 0 e 15 per la variabile del volume. Girate il programma per scoprire come, cambiando il volume, si cambia casualmente il suono.

Comporre Musica

L'istruzione SOUND può produrre anche toni musicali. La seguente scala comprende le note musicali e la loro tonalità:

Nota	Tonalità
alto C	29
B	31
A	35
G	40
F	45
E	47
D	53
C	60
B	64
A	72
G	81
F	91
E	96
D	108
mezzo C	121

Battete e fate girare il seguente programma:

```
NEW
10 REM ** SIMPLE SONG
15 DIM PITCH$ (1)
20 VOICE=0:PITCH=0:TONE=10:VOL=8
30 REM ** C=121:D=108:E=96:F=91
40 TRAP 300
50 PRINT "↖"
```

```
60 PRINT "NOTES FOR SIMPLE SONG"
65 FOR NOTE= 1 TO 8
70 READ PITCH
80 SOUND VOICE,PITCH,TONE,VOL
90 GOSUB 200
100 PRINT:PRINT PITCH$
110 FOR PAUSE=1 TO 500:NEXT PAUSE
120 SOUND 0,0,0,0
130 NEXT NOTE
140 GOTO 300
150 REM ** DATA FOR NOTES
160 DATA 121,121,108,96,96,91,108,121
200 REM ** PRINT NOTES
210 IF PITCH=121 THEN PITCH$="C"
220 IF PITCH=108 THEN PITCH$="D"
230 IF PITCH=96 THEN PITCH$="E"
240 IF PITCH=91 THEN PITCH$="F"
250 RETURN
300 PRINT: PRINT "END OF SIMPLE SONG":END
RUN
```

I comandi GOSUB-RETURN e READ-DATA permettono al computer di produrre note differenti, inserendo una serie di valori per la variabile PITCH. Il comando GOSUB indica al computer di eseguire la “subrutine” che incomincia alla riga 200 fino alla riga 250; il comando RETURN fa ritornare il computer alla linea immediatamente sotto la riga GOSUB. Il comando READ indica al computer di prendere un dato dalla riga DATA ed inserirlo nella variabile. Il computer continua a girare sul programma fino a che tutti i valori sulla riga DATA sono stati usati.

Il programma adopera anche un loop FOR-NEXT per determinare la durata di una nota. Usando differenti iterazioni di FOR-NEXT, provate a modificare il programma per produrre note intere, mezze note, ed altri tipi di note.

GRAFICHE A COLORI

Il vostro ATARI 130XE ha 16 modi grafici che comprendono 128 colori. Per cominciare, questa sezione presenta 6 differenti modi ed alcuni dei comandi grafici piú importanti.

La seguente lista contiene i 16 colori di base e i loro corrispondenti valori numerici (i colori possono variare a seconda di come è stato regolato il vostro televisore):

0 Grigio	8 Blue
1 Oro	9 Azzurro
2 Arancione	10 Turchese
3 Arancio-rosso	11 Verde-blue
4 Rosa	12 Verde
5 Viola	13 Giallo-verde
6 Rosso-arancione	14 Arancione-verde
7 Blue	15 Arancione chiaro

Gli altri 112 colori sono ottenuti aggiungendo luminosità, definita da un numero pari tra 0 e 14. Piú alto è il numero di luminosità, piú chiaro e brillante è il colore.

I registri del colore sono un altro importante elemento nella grafica ATARI. I registri del colore potrebbero essere considerati o visti come dei barattoli di pittura. Ogni registro può contenere uno qualsiasi dei 128 colori. Poiché ci sono cinque registri possono essere visualizzati un massimo di cinque colori. I cinque registri del colore sono numerati 0, 1, 2, 3 e 4.

SETCOLOR è uno dei principali comandi di grafica. Il formato del comando è SETCOLOR 2,10,8; al primo numero corrisponde il registro dei colori, al secondo il colore ed al terzo la luminosità.

Modo Grafico 0

I registri del colore si comportano in modo differente a seconda dei diversi modi grafici. Le funzioni dei registri, nel modo grafico 0 (modo di testo), sono mostrate qui sotto:

Colore non definito	Registro	Funzione
	0	Non usato
Azzurro	1	Luminosità del testo
Blue scuro	2	Fondo
	3	Non usato
Nero	4	Bordo

I colori non definiti sono colori che il computer usa automaticamente quando non gli vengono indicati altri colori. Per cambiare i colori, usate SETCOLOR e battete:

```
SETCOLOR 2, 3, 4
```

Quando premete [Return] lo schermo diventa arancione. La trasformazione del colore avviene perché nell'istruzione SETCOLOR, il 2 rappresenta il colore dello schermo, il 3 il colore arancione ed il 4 la luminosità. Cambiate il 4 in 6. Il colore arancio diventa piú chiaro. Cambiate il 6 in 7. Non succede niente perché solo i numeri

pari tra 0 e 14 definiscono la luminosità. Se battete un numero dispari il computer usa il colore corrispondente al numero precedente. Cambiate il 7 in 8 e notate come il colore diventa piú chiaro. Il seguente programma mostra tutti i 128 colori e gamma di luminosità:

```
NEW
10 REM ** 128 ATARI COLORS
20 REM ** 16 COLORS
30 FOR COLOR=0 TO 15
40 REM ** 8 LUMINANCES
50 FOR LUMINANCE=0 TO 14 STEP 2
60 SETCOLOR 2,COLOR,LUMINANCE
65 PRINT "COLOR=";COLOR;" LUMINANCE=";LUMINANCE
70 REM ** PAUSE TO SEE COLOR
80 FOR PAUSE=1 TO 600:NEXT PAUSE
90 NEXT LUMINANCE
100 NEXT COLOR
RUN
```

Quando la luminosità è 10 il testo non si vede piú; questo perché la luminosità non definita del testo è pure 10 (la luminosità non definita è la luminosità che il computer automaticamente usa quando non gli viene definito un'altra luminosità). Ogni qualvolta la luminosità del fondo è uguale a quella del testo, questo sembra scomparire. Fate attenzione alle luminosità del fondo e del testo quando lavorate con i colori e luminosità nel modo grafico 0, abbreviato GR.0. Battere GR.0 per ristabilire i colori normali sullo schermo.

Cambiate da SETCOLOR 2 a SETCOLOR 4 a linea 60 e fate girare il programma di nuovo. Poiché il registro 4 gestisce il bordo, questo cambia colore invece del fondo. Battere FR.0 per ristabilire i colori normali.

I Modi Grafici 1 e 2

I modi grafici 1 e 2 danno dei testi di grandi dimensioni e opzioni di colori. Il modo grafico 1 è identico al modo grafico 2, ad eccezione dei testi che sono di doppia altezza. Il modo 1 ha 24 linee orizzontali di schermo, il modo 2 ne ha 12. Per usare il modo grafico 1 battete:

```
NEW
10 GRAPHICS 1
20 PRINT #6;"GRAPHICS MODE ONE"
```

Fate girare il programma. Il testo grafico 1 arancione, si trova in alto allo schermo. In basso, c'è una fascia blu contenente la parola READY. La fascia blu è la finestra del testo e mostra il testo nel modo grafico 0. Battete GR.0 per ritornare al modo di testo.

Per visualizzare sullo schermo un testo a grandi dimensioni nei modi grafici 1 e 2, usa PRINT 6; seguito da virgolette e il testo che volete visualizzare. Questa espressione è una variante della istruzione PRINT che avete imparato prima.
Adesso listate il programma. Cambiate MODE in mode e fate girare il programma. MODE diventa verde. Battete LIST 20. Usando il tasto video inverso, cambiate mode alla linea 20 in MODE e fate girare il programma. MODE adesso è blu. Listate la linea di nuovo e cambiate MODE in mode e fate girare il programma. MODE adesso è rosso.

Battete e fate girare il seguente programma:

```
NEW
10 REM ** COLORFUL TEXT
20 GRAPHICS 1
30 PRINT #6;"ORANGE"
40 PRINT #6;"green"
50 PRINT #6;"DARK BLUE"
60 PRINT #6;"red"
70 PRINT "COLORFUL TEXT"
RUN
```

Come potete vedere, il modo grafico 1 è capace di visualizzare cinque colori contemporaneamente. Quattro colori di testo e uno di fondo. I colori possono essere cambiati usando SETCOLOR secondo le norme della seguente tabella.

Registro	Colore non definito	Tipo di carattere	Colore	LUM
0	Arancio	Maiuscolo	2	8
1	Verde chiaro	Minuscolo	12	10
2	Blu scuro	Maiuscolo inverso	9	4
3	Rosso	Minuscolo inverso	4	6
4	Nero	Fondo	0	0

Battete SETCOLOR 4,15,5. Registro 4 (il fondo) diventa un arancione rossastro. Adesso il testo in blu scuro è difficile da distinguere. Usa SETCOLOR per cambiarlo. Come da tabella, il registro 2 controlla il testo blu scuro. L'istruzione SETCOLOR 2,8,6 risolve il problema schiarendo il blu scuro del testo. Aggiungete queste linee al programma Testo Colorato:

```
100 FOR COLOR=0 TO 15
110 SETCOLOR 2,COLOR,8
120 FOR DELAY=1 TO 400:NEXT DELAY
130 NEXT COLOR
```

Fate girare il programma. La finestra del testo in basso allo schermo ed il testo blu scuro cambiano colore. Questo perché il registro 2 controlla sia la finestra del testo che la visualizzazione del testo.

Eliminare la Finestra del Testo

A volte non volete vedere la finestra del testo assieme ai vostri programmi. Per eliminare la finestra del testo aggiungete 16 al numero del modo grafico. Sostituite la linea 20 con l'istruzione GRAPHICS 17 e cancellate la linea 70. L'istruzione: PRINT visualizza sempre il modo grafico 0. Se state usando il modo grafico 1 o 2, se non avete una finestra di testo e usate l'istruzione PRINT e l'istruzione PRINT 6; il computer si confonderà e visualizzerà il tutto nel modo 0. Aggiungete questa linea:

```
70 PRINT "WINDOW TEST"
```

Fate girare il programma per vedere cosa succede. Se usate PRINT E PRINT#6; dovrete usare una finestra di testo per visualizzare sullo schermo il modo 1.

Cancellate le linee 100, 110, 120 e 130. Fate girare il programma. Le parole WINDOW TEST e poi READY si troveranno in alto allo schermo. Listate il programma. La linea 20 specifica modo 17 (modo 1 senza la finestra di testo), ma dove si trova? Sostituite la linea 70 con questa:

```
70 GOTO 70
```

Quando fate girare il programma, il modo 1 ritorna. Quando usate il modo 1 o 2 senza la finestra di testo, dovete inserire un loop GOTO per mantenere l'immagine sullo schermo altrimenti passerà troppo rapidamente per essere vista. Premendo il tasto [Break] si ritorna al modo 0.

Per vedere un esempio del modo 2, listate il programma TESTO COLORATO e sostituite la linea 20 con:

```
20 GRAPHICS 18
```

L'istruzione Graphics 18 significa il modo grafico 2 piú 16 (senza finestra di testo). Fate girare il programma. Adesso avete un testo colorato piú grande.

Per ristabilire i colori originali premete il tasto [Reset] o scrivere SETCOLOR 2, 9, 4. Nell'ATARI BASIC non perdete il programma quando premete il tasto [Reset]. Questo non potrebbe essere vero per gli altri linguaggi e programmi.

Modo Grafico 3

Il modo grafico 3 consiste in una griglia di 40 colonne e 24 righe (20 se usate la finestra di testo). Battete e fate girare il seguente programma:

```
NEW
10 GRAPHICS 3
20 COLOR 1
30 PLOT 0,0
RUN
```

Nell'angolo in alto a sinistra c'è un punto arancione. Il punto, o pixel, è una unità dello schermo grafico. L'istruzione COLOR determina il colore del pixel. Il numero dopo l'istruzione COLOR determina quale registro dei colori usare per il colore del pixel. L'istruzione COLOR non mette un colore nel registro; l'istruzione SETCOLOR ha questo compito. L'istruzione COLOR seleziona solo il registro da usare per posizionare i pixel, il quale prende il colore che si trova nel registro. Per rendere questo piú chiaro sostituite la linea 20 con:

```
20 COLOR 2
```

Fate girare il programma. Il pixel arancione diventa verde chiaro. Considerate ogni pixel come un carattere di testo. Nei modi 1 e 2, avete usato caratteri maiuscoli, minuscoli e video inverso per selezionare il colore del testo. Nel modo 3 usate l'istruzione COLOR per selezionare i colori del pixel.

PLOT: Tracciare i Punti sulla Griglia

PLOT è come l'istruzione PRINT #6; con la differenza che visualizza i pixel invece delle lettere e dei numeri. L'istruzione COLOR è simile al metodo di selezione del colore con maiuscole/minuscole/inverso. COLOR seleziona soltanto il registro. I colori definiti sono l'arancione, il verde chiaro, il blu scuro e il nero. Per cambiare i colori in qualsiasi registro, usate l'istruzione SETCOLOR.

I registri dei colori sono come quattro barattoli di pittura. SETCOLOR seleziona i colori con cui si devono riempire i barattoli, COLOR seleziona il barattolo dentro il quale sarà immerso il pennello. PLOT determina su quale punto dello schermo dovrà essere messo il pennello.

DRAWTO: Collegare i Punti.

Aggiungete questa linea:

```
40 DRAWTO 39,0
```

Fate girare il programma. Una linea verde chiaro attraversa la parte alta dello schermo. Dopo aver posizionato un pixel, usate l'istruzione DRAWTO per posizionare un secondo pixel e congiungerli con una linea. La linea 40 indica al computer di posizonare un pixel alla colonna 39, riga 0 e poi di congiungerli con una linea. Adesso battete:

```
DRAWTO 39,19
```

L'istruzione posiziona un pixel nell'angolo in basso a destra dello schermo appena sopra la finestra di testo e poi tira una linea per congiungere 39,0 e 39,19.
Adesso battete:

```
DRAWTO 0,19
```

Per completare il rettangolo, battete:

```
DRAWTO 0,0
```

Adesso battete GR.0 e listate il programma. Aggiungete queste linee:

```
50 DRAWTO 39,19
60 DRAWTO 0,19
70 DRAWTO 0,0
```

SETCOLOR e COLOR

Quando fate girare il programma, il computer disegna un rettangolo verde. Per ravvivare lo schermo battete:

```
35 COLOR 1
45 COLOR 2
55 COLOR 1
65 COLOR 3
```

Fate girare il programma e notate come il rettangolo cambia colore.

Per cambiare colore in un registro usate SETCOLOR. Potreste dedurre che COLOR 1 seleziona il colore del registro 1 e COLOR 2 seleziona il colore del registro 2.
Purtroppo questa conclusione non è totalmente vera. Il Modo 3 ha quattro registri e quattro colori, i registri sono numerati 0, 1, 2, e 4 mentre i colori sono numerati 0, 1, 2, e 3. Per semplificare le cose fate una tabella:

Colore 0 = Registro 4 Nero
Colore 1 = Registro 0 Arancione
Colore 2 = Registro 1 Verde chiaro
Colore 3 = Registro 2 Blu scuro

Battete GR.0, listate il programma e sostituite COLOR 2 alla linea 20 con COLOR 1. L'istruzione COLOR 1 seleziona il registro 0, l'arancione è il colore non definito del registro 0. Per cambiare colore nel registro 0, usate l'istruzione SETCOLOR.
Aggiungete la linea seguente:

```
15 SETCOLOR 0,4,6
```

Quando fate girare il programma, la linea arancione diventa di color rosa. Avete cambiato il colore delle linee usando l'istruzione SETCOLOR per cambiare la pittura nel barattolo (il colore nel registro), ma non l'istruzione COLOR per scegliere un altro barattolo (registro). La luminosità di colore del registro 0 cambia anche la luminosità del testo nella finestra di testo.
Adesso aggiungete:

```
42 SETCOLOR 1,2,8
```

Il verde chiaro alla destra del rettangolo diventa color oro.
Aggiungete:

```
62 SETCOLOR 2,11,4
```

Fate girare il programma. Non solo la parte sinistra del rettangolo diventa verde, ma anche la finestra del testo diventa verde. Perciò il registro 2 controlla anche il colore della finestra di testo.

Adesso dovreste essere capaci di usare le istruzioni SETCOLOR e COLOR per ottenere una grande varietà di colori e tinte nei vostri programmi.

Modi Grafici 5 e 7

La differenza tra i modi 3, 5 e 7 può essere illustrata molto facilmente. Sostituite la linea 10 con

```
10 GRAPHICS 5
```

Quando fate girare il programma appare un rettangolo ancora piú piccolo perché i pixel sono piú piccoli. Con la finestra di testo, la griglia nel modo 3 è composta da 39 colonne e 20 linee. La griglia nel modo 5 ha 80 colonne e 4 linee. Cambiate ora la linea 10 con

```
10 GRAPHICS 7
```

Quando fate girare il programma appare un rettangolo ancora piú piccolo. La griglia nel modo 7 ha 160 colonne e 80 linee.

Piú piccoli sono i pixel, piú alta è la definizione. Dei tre modi disponibili, il modo 3 ha la piú bassa definizione e il modo 7 la piú alta. Provate a disegnare un rettangolo vicino ai bordi dello schermo nei modi 5 e 7.

Il seguente programma illustra tutto quello che avete provato in questa sezione. Battetelo e fatelo girare:

```
NEW
5 REM ** BILL'S BOX (PLOT AND DRAW)
10 PRINT "WHICH MODE (3,5, OR 7)";
20 LEFT=0:TOP=0
30 INPUT MODE
40 IF MODE=3 THEN RIGHT=39:BOTTOM=19
50 IF MODE=5 THEN RIGHT=79:BOTTOM=39
60 IF MODE=7 THEN RIGHT=159:BOTTOM=79
70 GRAPHICS MODE
80 PRINT "      GRAPHICS MODE ";MODE
90 FOR COUNT=1 TO 1000
100 COLOR 2
110 TRAP 240
115 REM ** DRAW BOX
120 PLOT LEFT,TOP
130 COLOR 1
140 DRAWTO RIGHT,TOP
150 COLOR 2
160 DRAWTO RIGHT,BOTTOM
170 COLOR 1
180 DRAWTO LEFT,BOTTOM
190 COLOR 3
200 DRAWTO LEFT,TOP
205 REM ** DELAY LOOP
210 FOR DELAY=1 TO 500:NEXT DELAY
215 REM ** SIZE OF NEXT BOX
220 LEFT=LEFT+2:TOP=TOP+2:RIGHT=RIGHT-2:BOTTOM=BOTTOM-2
230 NEXT COUNT
240 PRINT "      THAT'S ALL FOLKS!"
250 END
```

Provate ad usare l'istruzione SETCOLOR per cambiare i colori nel programma Bill's Base.

Potete imparare molto di piú sulla grafica ATARI, anche come usare altri modi grafici e creare caratteri animati. Guardate l'Appendice F: Risorse per trovare libri, riviste e gruppi utenti che vi potranno aiutare ad esplorare meglio il mondo del BASIC ATARI e del vostro computer ATARI 130XE.

Parte 3
Appendici

A. Programmi Dimostrativi

Con un po' di immaginazione e buone tecniche di programmazione, il vostro Computer ATARI può compiere miracoli. Questi programmi dimostrativi vi mostreranno le versatilità del vostro ATARI 130XE e vi motiveranno a scrivere dei programmi da voi stessi.

Battete i programmi esattamente come sono stati stampati, premete [Return] alla fine di ogni linea. Quando avete finito, battete sulla tastiera la parola RUN, premete [Return] ed osservate come il Vostro Computer ATARI sembra animarsi.

Note: Quando la spaziatura in una linea di programma è importante, una nota alla fine del programma indicherà il numero esatto di spazi occorrenti.

IL CHOO-CHOO ATARI

Gli effetti sonori sono una particolarità di ATARI. Se chiudete gli occhi quando il CHOO-CHOO ATARI gira, vi sembrerà di essere sul Marrakesh Express.

```
10 POKE 765,255:POKE 580,1
20 GRAPHICS 17:POKE 712,148: POSITION 1,10:PRINT #6;
"THE ATARI CHOO-CHOO"
30 FOR X=15 TO 0 STEP -1-P:SOUND 1,0,0,X
40 R=INT(RND (0)*300)+1
50 IF R=30 THEN SOUND 3, 36, 10, 10: SOUND 2, 48, 10,
10:GOSUB 90: SOUND 3,0,0,0:SOUND 2, 0, 0, 0
60 NEXT X:P=P+0.03
70 IF P>=5 THEN P=5
80 GOTO 30
90 POKE 77,0:POSITION 8,12: PRINT #6; "toot":FOR A=1 TO
400: NEXT A:POSITION 8,12: PRINT #6;"    ":RETURN
```

Note: La linea 90 richiede quattro spazi vuoti tra le virgolette.

IL BIG BANG

Chiudete la porta prima di far girare questo programma per non disturbare i vostri vicini:

```
10 POKE 764,255:POKE 580, 1
20 GRAPHICS 17
30 FOR X=10 TO 100: SOUND 0, X, 10, 10:SOUND 1, X-2, 10, 8:
SOUND 2, X+2, 10, 12:NEXT X
40 SOUND 1, 0, 0, 0:SOUND 2, 0, 0, 0
50 POSITION 4,11: PRINT #6; "BAROOOOMMM!"
60 FOR DECAY=15 TO 0 STEP -0.5: FOR B=1 TO 20:
SOUND 0,100,B,DECAY: POKE 712,B:NEXT B:NEXT DECAY
70 GRAPHICS 1+32:POKE 712, 148
80 POKE 752,1:PRINT : PRINT " Press Start to set off
another explosion."
90 IF PEEK(53279)<>6 THEN GOTO 90
100 GOTO 20
```

METTERE IN ORDINE LE PAROLE

Questo programma mette le parole in ordine alfabetico. Sostituite le parole nelle espressioni DATA alle linee 10 e 20 per classificare le parole di vostra scelta. Ricordatevi di separare ogni vostra parola con una virgola.

```
10 DATA ATARI,DISK DRIVE, MONITOR,COMPUTER,TOUCH
TABLET,PRINTER,KEYBOARD
20 DATA SOFTWARE,PROGRAM RECORDER,WORD PROCESSING,
ACCOUNTING,DATA BASE,FUN
30 DIM Z$(1000),A(50),A$(20),S(10)
40 S(1)=1:FOR L=1 TO 9: S(L+1)=S(L)*3+1:NEXT L
50 TRAP 80:GRAPHICS 0:? "HERE IS THE LIST:"
60 READ A$:B=LEN(Z$):C=LEN(A$): Z$
(B+1,B+1)=CHR$(C):? A$
70 Z$(B+2,B+1+C)=A$: Q=Q+1:A(Q)=B+1:GOTO 60
80 ? :? "READY TO SORT...",:P=0
90 P=P+1:IF S(P+2)<Q THEN 90
100 FOR I=P TO 1 STEP -1:S=S(I): FOR J=S+1 TO
Q:L=J-S:A=A(J): B=A(L)
110 IF Z$(A+1,A+ASC(Z$(A,A)))>Z$ (B+1,B+ASC(Z$(B,B)))
THEN 130
120 A(L+S)=B:L=L-S:IF L>0 THEN B=A(L):GOTO 110
130 A(L+S)=A:NEXT J:NEXT I:? : ? "SORTED."
140 FOR L=1 TO Q:A=A(L): ? Z$(A+1,A+ASC(Z$(A,A))):NEXT L
```

GIOCATORI E MISSILI

Questo programma usa una tecnica chiamata Grafica del Missile Giocatore, per creare un mostro rosa che si muove sullo schermo di fronte ad una barra verticale blu.
Se volete che il mostro si muova dietro la barra blu, sostituite la linea di programma 150 con `150 POKE 623,4`.

```
10 POKE 764,255:POKE 580,1
20 GRAPHICS 3+16
30 FOR X=16 TO 24:FOR Y=0 TO 23: COLOR 3:PLOT X, Y:NEXT
Y:NEXT X
40 MEMTOP=PEEK(741)+256*PEEK(742)-1
50 PMBASE=INT((MEMTOP-1024) /1024) *1024
60 ADJTOP=PMBASE+384
70 POKE 742,INT (ADJTOP/256):
POKE 741,ADJTOP-256*PEEK(742)
80 POKE 54279,PMBASE/256
90 POKE 53277,2
100 POKE 559,34+8
110 P0=PMBASE+512
120 FOR A=P0 TO P0+128:POKE A,0: NEXT A
130 FOR A=P0+60 TO P0+67:READ B: POKE A,B:NEXT A
140 POKE 53256,3
150 POKE 623,1
160 POKE 704,108
170 POKE 53248,PEEK(20) :GOTO 170
180 DATA 60,126,129,153,255,36,66,129
```

TOPSY-TURVY

Quando fate girare Topsy-Turvy, il vostro schermo di riempirà di strane scritture. Per mettere ordine, premete [Start]. Per creare piú confusione, premete [Select].

```
10 POKE 764,255:POKE 580,1
20 GRAPHICS 18:POKE 712,128: POKE 755,5
30 POSITION 5,3:PRINT #6; "WELCOME TO"
40 POSITION 2,5:PRINT #6; "THE TOPSY-TURVY":POSITION
6,7:PRINT #6; "WORLD OF": POSITION 6,9
50 PRINT #6;"COMPUTERS"
60 IF PEEK(53279)=5 THEN POKE 755,5:POKE 712,128
70 IF PEEK(53279)=6 THEN POKE 755,1:POKE 712,99
80 GOTO 60
```

BATTERE UN MOTIVO SULLA TASTIERA

Questo programma assegna il valore musicale ai tasti della fila in alto alla tastiera. Premete solo un tasto per volta.

TASTO	VALORE MUSICALE
Insert	B
Clear	$B^{\flat}$ (or A#)
0	A
9	$A^{\flat}$ (or G#)
8	G
7	F#(or $G^{\flat}$)
6	F
5	E
4	$E^{\flat}$ (or D#)
3	D
2	$D^{\flat}$ (or C#)
1	C

```
10 DIM CHORD(37),TUNE(12)
20 GRAPHICS 0:? :? "      TYPE-A-TUNE PROGRAM"
25 ? :? "PRESS KEYS 1-9,0,<,> TO PRODUCE NOTES. ";
27 ?:? "RELEASE ONE KEY BEFORE PRESSING THE    NEXT."
28 ?:? "OTHERWISE, THERE MAY BE A DELAY."
30 FOR X=1 TO 37:READ A:CHORD(X)=A:NEXT X
40 FOR X=1 TO 12:READ A:TUNE(X)=A:NEXT X
50 OPEN #1,4,0, "K:"
55 OLDCHR=-1
60 A=PEEK(764):IF A=255 THEN 60
63 IF A=OLDCHR THEN 100
65 OLDCHR=A
70 FOR X=1 TO 12:IF TUNE(X)=A THEN SOUND
0,CHORD(X),10,8:GOTO 100
80 NEXT X
100 I=INT(PEEK(53775)/4):IF (I/2)=INT(I/2) THEN 60
110 POKE 764,255:SOUND 0,0,0,0:OLDCHR=-1 :GOTO 60
200 DATA 243,230,217,204,193,182,173,162,153,144,136,
128,121,114,108,102,96,91,85,81,76,72,68,64,60
210 DATA 57,53,50,47,45,42,40,37,35,33,31,29
220 DATA 31,30,26,24,29,27,51,53,48,50,54,55
```

Per suonare “Mary had a little Lamb” premete i seguenti tasti:

5, 3, 1, 3, 5, 5, 5   3, 3, 3   5, 8, 8   5, 3, 1, 3, 5, 5, 5   5, 3, 3, 5, 3, 1

Note: Nella linea 27 assicuratevi di inserire 3 spazi tra il THE e il NEXT.

ALTA MATEMATICA

Il vostro computer ATARI è un calcolatore fantasioso. Quando date due numeri qualsiasi al programma indicato qui sotto, il computer vi darà il massimo comune denominatore. Per esempio, se battete 690911 e 11214017 troverete immediatamente che il loro massimo comune denominatore è 53147.

```
10 ? CHR$(125):? "Enter two numbers. Press Return after
each entry."
20 INPUT N1,N2
30 GOSUB 90
40 ? "Their GCD is ";:? AN
50 POKE 752,1:POSITION 10,10: ? "Press Start to
continue."
60 IF PEEK(53279)<>6 THEN GOTO 60
70 POKE 752,0:? CHR$(125) :GOTO 10
80 REM ****SUBROUTINE****
90 AN=0:POKE 195,0:TRAP 130: M=(N1>=N2)*N1+(N2>N1)*N2:
N=(M=N1)*N2+(M=N2)*N1
100 IF INT(N1)<>N1 OR INT (N2)<>N2 THEN RETURN
110 P=M-INT (M/N)*N:M=N:N=P
120 IF P<>0 THEN GOTO 110
130 AN=M*(PEEK(195)=0):RETURN
```

COMPUTER BLUES

Il programma genera note musicali per comporre delle melodie molto interessanti per la viola programmata.

```
1 GRAPHICS 0:? :? "   COMPUTER BLUES":?
2 PTR=1
3 THNOT=1
5 CHORD=1
6 PRINT "BASS TEMPO (1=FAST)";
7 INPUT TEMPO
8 GRAPHICS 2+16:GOSUB 2000
10 DIM BASE(3,4)
20 DIM LOW(3)
25 DIM LINE(16)
26 DIM JAM(3,7)
30 FOR X=1 TO 3
40 FOR Y=1 TO 4
50 READ A:BASE(X,Y)=A
60 NEXT Y
70 NEXT X
80 FOR X=1 TO 3:READ A:LOW(X)=A
90 NEXT X
95 FOR X=1 TO 16:READ A:LINE(X)=A:NEXT X
96 FOR X=1 TO 3
97 FOR Y=1 TO 7
98 READ A:JAM(X,Y)=A:NEXT Y:NEXT X
100 GOSUB 500
110 T=T+1
115 GOSUB 200
120 GOTO 100
200 REM PROCESS HIGH STUFF
205 IF RND(0)<0.25 THEN RETURN
210 IF RND(0)<0.5 THEN 250
220 NT=NT+1
230 IF NT>7 THEN NT=7
240 GOTO 260
250 NT=NT-1
255 IF NT<1 THEN NT=1
260 SOUND 2,JAM(CHORD,NT),10,NT*2
280 RETURN
500 REM PROCESS BASE STUFF
510 IF BASS=1 THEN 700
520 BDUR=BDUR+1
530 IF BDUR<>TEMPO THEN 535
531 BASS=1:BDUR=0
535 SOUND 0,LOW(CHORD),10,4
540 SOUND 1,BASE(CHORD,THNOT),10,4
```

```
550 RETURN
700 SOUND 0,0,0,0
710 SOUND 1,0,0,0
720 BDUR=BDUR+1
730 IF BDUR<>1 THEN 800
740 BDUR=0:BASS=0
750 THNOT=THNOT+1
760 IF THNOT<>5 THEN 800
765 THNOT=1
770 PTR=PTR+1
780 IF PTR=17 THEN PTR=1
790 CHORD=LINE(PTR)
800 RETURN
1000 DATA 162,144,136,144,121,108,102,108,108,96,91,96
1010 DATA 243,182,162
1020 DATA 1,1,1,1,2,2,2,2,1,1,1,1,3,2,1,1
1030 DATA 60,50,47,42,40,33,29
1040 DATA 60,50,45,42,40,33,29
1050 DATA 81,68,64,57,53,45,40
2000 PRINT #6:PRINT #6:PRINT #6
2005 PRINT #6;"   Computer"
2006 PRINT #6
2010 PRINT #6;"    Blues"
2030 RETURN
```

LA BANDIERA DEGLI STATI UNITI

Questo programma prevede la suddivisione di colori per creare delle strisce. Usa il modo grafico 7 piú il 16, cosí che l'immagine appare a schermo pieno. Notate la corrispondenza delle istruzioni COLOR con le indicazioni SETCOLOR. Per aumentare il divertimento e la sperimentazione, aggiungete un'istruzione SOUND ed usate la combinazione READ/DATA da aggiungere dopo la linea 470 "The Star Splanged Banner".

```
10 REM DRAW THE UNITED STATES FLAG
20 REM HIGH RESOLUTION 4-COLOR GRAPHICS, NO TEXT WINDOW
30 GRAPHICS 7+16
40 REM SETCOLOR 0 CORRESPONDS TO COLOR 1
50 SETCOLOR 0,4,4:RED=1
60 REM SETCOLOR 1 CORRESPONDS TO COLOR 2
70 SETCOLOR 1,0,14:WHITE=2
80 REM SETCOLOR 2 CORRESPONDS TO COLOR 3
90 BLUE=3:REM DEFAULTS TO BLUE
100 REM DRAW 13 RED & WHITE STRIPES
110 C=RED
120 FOR I=0 TO 12
130 COLOR C
140 REM EACH STRIPE HAS SEVERAL HORIZONTAL LINES
150 FOR J=0 TO 6
160 PLOT 0,I*7+J
170 DRAWTO 159,I*7+J
180 NEXT J
190 REM SWITCH COLORS
200 C=C+1:IF C>WHITE THEN C=RED
210 NEXT I
300 REM DRAW BLUE RECTANGLE
310 COLOR BLUE
320 FOR I=0 TO 48
330 PLOT 0,I
340 DRAWTO 79,I
350 NEXT I
360 REM DRAW 9 ROWS OF WHITE STARS
370 COLOR WHITE
380 K=0:REM START WITH ROW OF 6 STARS
390 FOR I=0 TO 8
395 Y=4+I*5
400 FOR J=0 TO 4:REM 5 STARS IN A ROW
410 X=K+5+J*14:GOSUB 1000
420 NEXT J
430 IF K<>0 THEN K=0:GOTO 470
```

```
440 REM ADD 6TH STAR EVERY OTHER LINE
450 X=5+5*14:GOSUB 1000
460 K=7
470 NEXT I
500 REM IF KEY HIT THEN STOP
510 IF PEEK(764)=255 THEN 510
515 REM OPEN TEXT WINDOW WITHOUT CLEARING SCREEN
520 GRAPHICS 7+32
525 REM CHANGE COLORS BACK
530 SETCOLOR 0,4,4:SETCOLOR 1,0,14
550 STOP
1000 REM DRAW 1 STAR CENTERED AT X,Y
1010 PLOT X-1,Y:DRAWTO X+1,Y
1020 PLOT X,Y-1:PLOT X,Y+1
1030 RETURN
```

IGPAY ATINLAY

Questo programma converte parole o frasi in latino maccheronico. Fate attenzione a non usare parole di una sola lettera come A o I.

```
10 DIM A$(256):S=2
20 ? "Type in a word or sentence. Please don't exceed
three lines of text."
30 INPUT A$
40 FOR X=1 TO LEN(A$)
50 IF A$(X,X)=CHR$(32) THEN PRINT
A$(S,X-1);A$(S-1,S-1);"AY";" ";:S=X+2
60 IF X=LEN(A$) THEN PRINT A$(S,X); A$(S-1,S-1); "AY"
70 NEXT X
80 ? :? :? "THAT'S ALL FOLKS!"
```

GRAPHEEK

Battete quanto segue ed osservate:

```
10 DIM A$(35)
20 GRAPHICS 1
25 TRAP 90
30 A$="THIS IS A GRAPHICS  DEMONSTRATION."
40 FOR I=1 TO 33:? #6;A$(I,I);
50 S=PEEK(53770)
60 SOUND 0,S,10,14
70 FOR DELAY=0 TO 100:NEXT DELAY
80 NEXT I
90 SOUND 0,0,0,0:END
```

Note: Assicuratevi di avere inserito due spazi tra GRAPHICS e DEMONSTRATION nella linea 30.

ESREVER

Il titolo di questo programma è semplicemente la parola REVERSE stampata al rovescio. Per visualizzare parole scritte al rovescio, battete questo programma. Dopo averlo fatto girare la prima volta, un punto interrogativo apparirà sullo schermo. Battete sulla tastiera una parola od una frase e lasciate che il vostro ATARI 130XE faccia il resto.

```
10 DIM A$(180)
20 PRINT "Enter a word or short sentence and press
Return."
30 INPUT A$
40 FOR X=LEN(A$) TO 1 STEP -1
50 PRINT A$(X,X);
60 NEXT X
70 PRINT :PRINT :GOTO 20
```

PER PROTEGGERE I VOSTRI PROGRAMMI

Vi siete mai domandati come potete proteggere i vostri programmi dai curiosi? Un paio di trucchi vi potrebbero aiutare a tenere distanti i ficcanaso.

In primo luogo, battete questo programma:

```
10 FOR X=1 TO 50:POKE 710,X: NEXT X:GOTO 10
```

Aggiungete un'altra riga per disabilitare il tasto [Break]. Questa riga impedisce a chiunque di entrare nel programma e listarlo mentre sta girando. Questo serve anche per evitare di bloccare il programma nel caso che voi sbagliate a premere il tasto [Break] durante l'immissione di dati.

Cancellate GOTO 10 dal programma a colori e aggiungete questa linea:

```
20 POKE 16,64:POKE 53774,64:GOTO 10
```

Adesso fate girare il programma e cercate di arrestarlo premendo il tasto [Break]. Non ci riuscirete.

Per essere efficaci, le espressioni POKE devono essere inserite nel programma dopo ogni comando di modo grafico.

Disabilitare il tasto [Break] crea dei problemi. Alcuni programmatori esperti penseranno di poter entrare nel vostro programma premendo il tasto [Reset]. Per ingannare i curiosi, aggiungete questa linea:

```
5 POKE 580,1
```

A questo punto, quando l'intruso premerà il tasto [Reset], il programma dai colori lampeggianti purgherà la memoria del computer: se non ci sono programmi, non c'è niente da listare. L'espressione POKE deve sempre trovarsi al principio del programma.

IL GABBIANO SULL'OCEANO

Questo programma combina suoni e grafica. I suoni non sono suoni puri, ma simulano il mugghiare dell'oceano e lo stridio dei gabbiani. Per avere i simboli a linea 20, usare [Control] [G], [Control] [F], [Control] [R], [Control] [R].

```
10 DIM BIRD$(4)
20 BIRD$="\/--"
30 FLAG=1:ROW=10:COL=10
40 GRAPHICS 1:POKE 756,226:POKE 752,1
50 SETCOLOR 0,0,0:SETCOLOR 1,8,14
60 PRINT #6;"     the ocean"
70 R=INT(RND(0)*11)
80 POSITION 17,17
90 FOR T=0 TO 10
100 SOUND 0,T,8,4
110 FOR A=1 TO 50:NEXT A
120 IF RND(0)>0.8 THEN FOR D=10 TO 5 STEP -1:SOUND
1,0,10,INT(RND(0)*10):NEXT D: SOUND 1,0,0,0
130 GOSUB 200
140 NEXT T
150 FOR T=10 TO 0 STEP -1
160 SOUND 0,T,8,4
170 FOR A=1 TO 50:NEXT A
175 IF RND(0)>0.8 THEN FOR D=10 TO 5 STEP
-1:SOUND 1,D,10,8:NEXT D:SOUND 1,0,0,0
180 FOR H=1 TO 10:NEXT H
185 GOSUB 200
190 NEXT T
195 GOTO 70
200 GOSUB 300
210 POSITION COL,ROW
220 PRINT #6;BIRD$(FLAG,FLAG+1)
230 FLAG=FLAG+2:IF FLAG=5 THEN FLAG=1
240 RETURN
300 IF RND(0)>0.5 THEN RETURN
310 POSITION COL,ROW
320 PRINT #6;"  "
330 A=INT(RND(0)*3)-1
340 B=INT(RND(0)*3)-1
350 ROW=ROW+A
360 IF ROW=0 THEN ROW=1
370 IF ROW=20 THEN ROW=19
380 COL=COL+B
390 IF COL=0 THEN COL=1
400 IF COL>18 THEN COL=18
410 RETURN
```

Note: Tra le virgolette della linea 320 devono sempre essere inseriti due spazi.

ARTE CINETICA

Il programma nuove i colori per creare un arcobaleno di linee in continuo movimento.

```
10 REM KINETIC ART BY NEIL HARRIS
20 GRAPHICS 10
30 DIM A(3,50)
35 FOR L=0 TO 3:FOR M=0 TO 50:A(L,M)=0:NEXT M:NEXT L
40 HUE=INT(RND(1)*8+1):POKE
704+HUE,INT(RND(1)*8)*16+INT(RND(1)*4+4)
50 X1=INT(RND(1)*80):X2=INT(RND(1)*80)
:Y1=INT(RND(1)*192):Y2=INT(RND(1)*192)
60 COLOR 0:PLOT A(0,WHICH),A(1,WHICH):DRAWTO
A(2,WHICH),A(3,WHICH)
70 BOUNCE=BOUNCE-1:IF BOUNCE>0 THEN 90
80 BOUNCE=INT(RND(1)*10+10):BX1=INT(RND(1)*9-4)
:BX2=INT(RND(1)*9-4):BY1=INT(RND(1)*13-6)
:BY2=INT(RND(1)*13-6)
90 CHANGE=CHANGE-1:IF CHANGE>0 THEN 110
100 CHANGE=INT(RND(1)*10+5):HUE=INT(RND(1)*8+1):POKE
704+HUE,INT(RND(1)*256)
110 COLOR HUE:PLOT X1,Y1:DRAWTO X2,Y2
120 A(0,WHICH)=X1:A(1,WHICH)=Y1:A(2,WHICH)=X2:A
(3,WHICH)=Y2
130 WHICH=WHICH+1:IF WHICH>50 THEN WHICH=0
140 X1=X1+BX1:IF X1<0 OR X1>79 THEN BX1=-BX1:GOTO 140
150 X2=X2+BX2:IF X2<0 OR X2>79 THEN BX2=-BX2:GOTO 150
160 Y1=Y1+BY1:IF Y1<0 OR Y1>191 THEN BY1=-BY1:GOTO 160
170 Y2=Y2+BY2:IF Y2<0 OR Y2>191 THEN BY2=-BY2:GOTO 170
180 GOTO 60
```

B. Parole Riservate al BASIC

Nota: Il punto è obbligatorio dopo tutte le abbreviazioni delle parole principali.

PAROLA	ABBREVIAZIONE	BREVE DESCRIZIONE DELL'ESPRESSIONE IN BASIC
ABS		Dà il valore assoluto (senza segno) della variabile o espressione.
ADR		Dà l'indirizzo di memoria di una variabile stringa.
AND		Funziona come un'operatore logico; l'espressione è vera solo se ambedue le espressioni unite dall'AND sono vere.
ASC		Dà il valore numerico di un singolo carattere stringa.
ATN		Dà la tangente di un numero o espressione in radiani o gradi.
BYE	B.	Esce dal BASIC e ritorna al sistema operativo residente o processore della tastiera.
CLOAD	CLOA.	Carica i dati dal registratore di programmi nella RAM.
CHR$		Dà una singola stringa byte equivalente ad un valore numerico tra 0 e 255 nel codice ATASCII.
CLOG		Dà il logoritmo in base 10 di una espressione.
CLOSE	CL.	Chiude in un archivio alla fine delle operazioni I/O. Funziona come un comando I/O.
CLR		Esegue la funzione opposta di DIM: toglie le dimensioni alle stringhe e alle matrici.

COLOR	C.	Sceglie il registro del colore che viene usato per la grafica a colori.
COM		Esegue la stessa funzione di DIM.
CONT	CON.	Vuol dire "continuare". Fa ripartire il programma alla linea seguente dopo che è stato fermato da [Break] o dall'istruzione STOP.
COS		Dà il coseno di una variabile o espressione in gradi o radiani.
CSAVE		Trasferisce i dati dalla RAM al registratore di programma per conservazione su nastro.
DATA	D.	Come parte della combinazione READ-DATA, identifica gli elementi successivi (che devono essere separati da una virgola) come elementi di dati individuali.
DEG	DE.	Ordina al computer di eseguire funzioni trigonometriche in gradi invece di radioni (in mancanza di precisazione, lavora in radioni).
DIM	DI.	Riserva una parte specifica della memoria per variabili di matrici, serie e stringhe (tutte le variabili di matrici, serie e stringhe devono essere dimensionata con una espressione DIM).
DOS	DO.	Significa "Disk Operating System". Visualizza il menu (vedi il manuale DOS).
DRAWTO	DR.	Traccia una retta tra un punto di un grafico ed un punto specifico.
END		Arresta l'esecuzione del programma, chiude gli archivi (files); toglie i suoni. Può essere usato più di una volta in un programma (CONT. può essere usato per far ripartire il programma).

ENTER	E.	Memorizza dati o programmi in una forma priva di indicazioni (fonte). Funziona come un comando I/O.
EXP		Dà "e" (2.7182818) elevato alla potenza specificata.
FOR	F.	Viene usato con NEXT per impostare l'iterazione (loop) FOR-NEXT. Precede il campo (inizio, numero di esecuzione, fine) in cui la variabile d'iterazione opererà durante l'esecuzione dell'iterazione stessa.
FRE		Dà la quantità di memoria utente ancora disponibile.
GET	GE.	Usato principalmente con i dischetti per inserire un simbolo byte di dati.
GOSUB	GOS.	Salta al numero di riga indicato da dove inizia una subroutine.
GOTO	G.	Salta incondizionalmente al numero di riga indicato.
GRAPHICS	GR.	Specifica uno degli otto modi grafici disponibili (GR.O può essere usato per cancellare lo schermo).
IF		Causa un salto condizionale o l'esecuzione di un'altra espressione sulla medesima riga (soltanto se la prima espressione è vera).
INPUT	I.	Induce il computer a chiedere dati d'ingresso dalla tastiera. L'esecuzione continuerà solo quando il tasto Return è stato premuto e i dati immessi.
INT		Dà l'intero minore e piú vicino ad un valore specificato (l'arrotondamento è sempre verso il minore anche per i numeri negativi).
LEN		Dà la lunghezza in bytes o caratteri della stringa specificata (un byte contiene un carattere).

LET	LE.	Assegna un valore a un nome specifico di una variabile (LET è opzionale nel BASIC ATARI e può essere omesso).
LIST	L.	Visualizza o trasferisce all'esterno le righe di programma.
LOAD	LO.	Trasferisce nel computer i dati contenuti sul dischetto.
LOCATE	LOC.	Memorizza in una variabile specifica il valore che controlla un punto grafico specifico.
LOG		Dà il logaritmo naturale di un numero.
LPRINT	LP.	Ordina alla stampante di stampare un testo specifico.
NEW		Cancella tutto il contenuto della RAM di utente.
NEXT	N.	Genera un'iterazione FOR-NEXT per terminare o continuare, a seconda delle particolari variabili o espressioni (tutte le iterazioni – loop – sono eseguite almeno una volta).
NOT		Dà un 1 se l'espressione è falsa; dà uno 0 se è vera.
NOTE	NO.	Usato solamente con i dischetti (vedere il manuale DOS).
ON		Viene usato con GOTO o GOSUB per saltare al numero di riga (sono possibili salti multipli e differenti numeri di righe; dipendono dal valore della variabile o espressione ON.).
OPEN	O.	Apre l'archivio (file) specificato per operazioni di input/output.
OR		Viene usato come operatore logico tra due espressioni. Se una delle due è vera dà 1; se ambedue sono false dà 0.

PADDLE		Dà la posizione del controllore di gioco sulla manopola.
PEEK		Dà il contenuto in forma decimale del byte all'indirizzo di memoria specificato (RAM o ROM).
PLOT	PL.	Inserisce un singolo punto nella locazione X, Y specificata.
POINT	P.	Usato soltanto con i dischetti.
POKE	POK.	Memorizza il byte specificato nella locazione di memoria specificata (può essere usato solo con la RAM. Se tentate l'istruzione POKE con la ROM ottenete un Messaggio d'Errore).
POP		Toglie la variabile d'iterazione (loop) dalla catasta GOSUB. Usata quando viene fatta una deviazione anomala dall'iterazione.
POSITION	POS.	Posiziona il cursore in una posizione specifica dello schermo.
PRINT	PR. oppure?	Trasferisce i dati dal computer ad una specificata unità periferica. Funziona come un comando I/O.
PTRIG		Dà la situazione sul pulsante dei comandi per giochi.
PUT	PU.	Trasferisce un singolo byte dal computer ad una unità periferica specificata.
RAD		Ordina al computer di dare le informazioni delle funzioni trigonometriche in radiani invece di gradi (in mancanza di istruzioni il computer usa radiani – vedere DEG).
READ	REA.	Legge gli elementi della lista DATA e li assegna a variabili specifiche.

REM	R.	Significa "remarks" (notare). Non fa niente, ma permette l'inserimento di spiegazioni e commenti sulle righe di programma per futuro riferimento del programmatore.
RESTORE	RES.	Permette la lettura degli stessi dati più di una volta.
RETURN	RET.	Fà saltare il computer da una subroutine all'istruzione immediatamente successiva al GOSUB eseguita per ultima.
RND		Genera dei numeri casuali tra 0 e 1, ma mai 1.
RUN	RU.	Esegue il programma; predispone le variabili normali a 0; disidimensiona serie e stringhe.
SAVE	S.	Memorizza i dati e i programmi su dischetto sotto il nome di archivio (file) che accompagna l'istruzione SAVE. Funziona come un'istruzione I/O.
SETCOLOR	SE.	Memorizza la tinta e la luminosità di un colore in un particolare registro del colore.
SGN		Dà +1 se il valore è positivo, 0 se è zero, e −1 se è negativo.
SIN		Dà il segno trigonometrico in radiani o gradi di un determinato valore.
SOUND	SO.	Controlla il registro, l'altezza, la distorsione e il volume di un tono o nota.
SQR		Dà la radice quadrata di un determinato valore.
STATUS	ST.	Chiama la routine di status (condizione) di una periferica.

STEP		Usato con FOR-NEXT. Determina la quantità che deve essere saltata per ogni coppia di valori nella variabile d'iterazione.
STICK		Dà la posizione del joystick.
STRING		Dà 1 se il bottone del joystick non è premuto, 0 se è premuto.
STOP	STO.	Ferma il programma ma non chiude né gli archivi (files) nè il suono.
STR$		Dà il carattere di stringa uguale al valore numerico dato (per esempio STR$(65) dà 65 come una stringa).
THEN		Viene usato con IF. Se l'espressione è vera, allora l'istruzione THEN viene eseguita. Se l'espressione è falsa, il controllo del programma passa alla linea successiva.
TO		Viene usato con FOR, come "FOR X=1 TO 10." separa le espressioni del campo iterativo (loop).
TRAP	T.	Assume il controllo del programma nel caso di un errore di INPUT e dirige l'esecuzione ad uno specifico numero di linea.
USR		Dà il risultato di una subroutine in linguaggio macchina.
VAL		Dà il valore numerico equivalente di una stringa.
XIO	X.	Viene usata con i dischetti (vedere il manuale DOS) e con la grafica. Generalmente funziona come un'istruzione I/O.

C. Serie di Caratteri ATASCII

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
0	0		Control ,	á
1	1		Control A	ù
2	2		Control B	Ñ
3	3		Control C	É
4	4		Control D	Ç
5	5		Control E	ô
6	6		Control F	ò
7	7		Control G	ì
8	8		Control H	£
9	9		Control I	ï
10	A		Control J	ü
11	B		Control K	ä
12	C		Control L	Ö
13	D		Control M	ú
14	E		Control N	ó
15	F		Control O	ö
16	10		Control P	Ü
17	11		Control Q	â

Note:

1 . ATASCI vuol dire ATARI ASCII. Le lettere ed i numeri hanno gli stessi valori di quelli nell'ASCII, ma alcuni caratteri speciali sono differenti;

2 . A parte le eccezioni mostrate, i caratteri da 128 a 255 hanno i colori inversi dei caratteri da 1 a 127;

3 . Sommate 32 al codice delle maiuscole per ottenere il codice della medesima lettera minuscola;

4 . Per ottenere il codice ATASCII, istruite il computer (modo diretto) con PRINT ASC ("______________"). Riempite gli spazi vuoti con una lettera o un carattere. Dovete usare le virgolette!

5 . Nella normale visualizzazione, i tasti sono mostrati con simboli bianchi su fondo nero; i simboli inversi dei tasti sono mostrati con simboli neri su fondo bianco.

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
18	12	─	Control R	û
19	13	┼	Control S	î
20	14	●	Control T	é
21	15	▄	Control U	è
22	16	▏	Control V	ñ
23	17	┬	Control W	ê
24	18	┴	Control X	å
25	19	▌	Control Y	à
26	1A	└	Control Z	Å
27	1B	␛	Esc Esc	
28	1C	↑	Esc Control -	
29	1D	↓	Esc Control =	
30	1E	←	Esc Control +	
31	1F	→	Esc Control *	
32	20		Space bar	
33	21	!	Shift 1	
34	22	"	Shift 2	
35	23	#	Shift 3	
36	24	$	Shift 4	
37	25	%	Shift 5	
38	26	&	Shift 6	
39	27	'	Shift 7	
40	28	(	Shift 9	
41	29	)	Shift 0	
42	2A	*	*	
43	2B	+	+	
44	2C	,	,	
45	2D	-	-	
46	2E	.	.	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
47	2F	/	/	
48	30	0	0	
49	31	1	1	
50	32	2	2	
51	33	3	3	
52	34	4	4	
53	35	5	5	
54	36	6	6	
55	37	7	7	
56	38	8	8	
57	39	9	9	
58	3A	:	Shift ;	
59	3B	;	;	
60	3C	<	<	
61	3D	=	=	
62	3E	>	>	
63	3F	?	Shift /	
64	40	@	Shift 8	
65	41	A	A	
66	42	B	B	
67	43	C	C	
68	44	D	D	
69	45	E	E	
70	46	F	F	
71	47	G	G	
72	48	H	H	
73	49	I	I	
74	4A	J	J	
75	4B	K	K	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
76	4C	L	L	
77	4D	M	M	
78	4E	N	N	
79	4F	O	O	
80	50	P	P	
81	51	Q	Q	
82	52	R	R	
83	53	S	S	
84	54	T	T	
85	55	U	U	
86	56	V	V	
87	57	W	W	
88	58	X	X	
89	59	Y	Y	
90	5A	Z	Z	
91	5B	[	Shift ,	
92	5C	\	Shift +	
93	5D	]	Shift .	
94	5E	^	Shift *	
95	5F	_	Shift -	
96	60	♦	Control .	i
97	61	a	a	
98	62	b	b	
99	63	c	c	
100	64	d	d	
101	65	e	e	
102	66	f	f	
103	67	g	g	
104	68	h	h	
105	69	i	i	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
106	6A	j	j	
107	6B	k	k	
108	6C	l	l	
109	6D	m	m	
110	6E	n	n	
111	6F	o	o	
112	70	p	p	
113	71	q	q	
114	72	r	r	
115	73	s	s	
116	74	t	t	
117	75	u	u	
118	76	v	v	
119	77	w	w	
120	78	x	x	
121	79	y	y	
122	7A	z	z	
123	7B		Control ;	Ä
124	7C			Shift =	
125	7D		Esc Control < or Esc Shift <	
126	7E		Esc Delete Bk Sp	
127	7F		Esc Tab	
128	80		Control ,	
129	81		Control A	
130	82		Control B	
131	83		Control C	
132	84		Control D	
133	85		Control E	
134	86		Control F	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
135	87		Control G	
136	88		Control H	
137	89		Control I	
138	8A		Control J	
139	8B		Control K	
140	8C		Control L	
141	8D		Control M	
142	8E		Control N	
143	8F		Control O	
144	90		Control P	
145	91		Control Q	
146	92		Control R	
147	93		Control S	
148	94		Control T	
149	95		Control U	
150	96		Control V	
151	97		Control W	
152	98		Control X	
153	99		Control Y	
154	9A		Control Z	
155	9B	EOL	Return	
156	9C		Esc Shift Delete Bk Sp	
157	9D		Esc Shift >	
158	9E		Esc Control Tab	
159	9F		Esc Shift Tab	
160	A0		Space bar	
161	A1		Shift 1	
162	A2		Shift 2	
163	A3		Shift 3	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
164	A4	$	◩ Shift 4	
165	A5	%	◩ Shift 5	
166	A6	&	◩ Shift 6	
167	A7	'	◩ Shift 7	
168	A8	(	◩ Shift 9	
169	A9	)	◩ Shift 0	
170	AA	*	◩ *	
171	AB	+	◩ +	
172	AC	,	◩ ,	
173	AD	-	◩ -	
174	AE	.	◩ .	
175	AF	/	◩ /	
176	B0	0	◩ 0	
177	B1	1	◩ 1	
178	B2	2	◩ 2	
179	B3	3	◩ 3	
180	B4	4	◩ 4	
181	B5	5	◩ 5	
182	B6	6	◩ 6	
183	B7	7	◩ 7	
184	B8	8	◩ 8	
185	B9	9	◩ 9	
186	BA	:	◩ Shift ;	
187	BB	;	◩ ;	
188	BC	<	◩ <	
189	BD	=	◩ =	
190	BE	>	◩ >	
191	BF	?	◩ Shift /	
192	C0	@	◩ Shift 8	
193	C1	A	◩ A	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
194	C2	B	◩ B	
195	C3	C	◩ C	
196	C4	D	◩ D	
197	C5	E	◩ E	
198	C6	F	◩ F	
199	C7	G	◩ G	
200	C8	H	◩ H	
201	C9	I	◩ I	
202	CA	J	◩ J	
203	CB	K	◩ K	
204	CC	L	◩ L	
205	CD	M	◩ M	
206	CE	N	◩ N	
207	CF	O	◩ O	
208	D0	P	◩ P	
209	D1	Q	◩ Q	
210	D2	R	◩ R	
211	D3	S	◩ S	
212	D4	T	◩ T	
213	D5	U	◩ U	
214	D6	V	◩ V	
215	D7	W	◩ W	
216	D8	X	◩ X	
217	D9	Y	◩ Y	
218	DA	Z	◩ Z	
219	DB	[	◩ Shift ,	
220	DC	\	◩ Shift +	
221	DD	]	◩ Shift .	
222	DE	^	◩ Shift *	
223	DF	_	◩ Shift -	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
224	E0	♦	◩ Control	
225	E1	a	◩ a	
226	E2	b	◩ b	
227	E3	c	◩ c	
228	E4	d	◩ d	
229	E5	e	◩ e	
230	E6	f	◩ f	
231	E7	g	◩ g	
232	E8	h	◩ h	
233	E9	i	◩ i	
234	EA	j	◩ j	
235	EB	k	◩ k	
236	EC	l	◩ l	
237	ED	m	◩ m	
238	EE	n	◩ n	
239	EF	o	◩ o	
240	F0	p	◩ p	
241	F1	q	◩ q	
242	F2	r	◩ r	
243	F3	s	◩ s	
244	F4	t	◩ t	
245	F5	u	◩ u	
246	F6	v	◩ v	
247	F7	w	◩ w	
248	F8	x	◩ x	
249	F9	y	◩ y	
250	FA	z	◩ z	
251	FB	♣	◩ Control ;	
252	FC	|	◩ Shift =	
253	FD	↖	Esc Control 2	

Codice Decimale	Codice Esadecimale	Carattere ATASCII	Battute di Tastiera	Caratteri Europei
254	FE		Esc Control Delete Bk Sp	
255	FF		Esc Control >	

D. Messaggi di Errore

NUMERO DI CODICE DELL'ERRORE	CODICE DEL MESSAGGIO D'ERRORE
2	**Insuffient Memory** (Memoria insufficiente): Non c'è abbastanza memoria di RAM disponibile per memorizzare l'espressione o il nome delle variabili o per dimensionare la variabile della nuova stringa.
3	**Value Error** (Errore di valore): Un valore che doveva essere positivo intero è invece negativo; il valore non è compreso nel campo specifico.
4	**Too Many Variables** (Troppe Variabili): Il limite massimo di 128 nomi di variabili è stato superato.
5	**Stringa Length Error** (Errore della Lunghezza di Stringa): L'utente ha cercato di memorizzare variabili stringa che superano la lunghezza di stringa dimensionata.
6	**Out of Data** (Mancanza di Dati): L'istruzione READ richiede piú dati di quelli forniti dall'istruzione DATA.
7	**Line Number Greater than 32767** (Numero di linea maggiore di 32767): Il numero di linea usato è maggiore di 32767.
8	**Input Statement Error** (Errore nell'Espressione di Entrata): L'utente ha cercato di usare un valore non numerico con una variabile numerica.
9	**Array or String DIM Error** (Errori DIM di Serie e Stringhe): La grandezza supera 5460 per le serie di numeri o 32767 per le stringhe; una serie o stringa è stata ridimensionata; un riferimento è stato fatto ad una serie o stringa non dimensionata.
11	**Floating Point Overflow/Underflow** (Eccesso/Difetto del Punto Fluttuante): L'utente ha cercato di dividere per zero o di usare un numero piú grande di 1×10^{98} o piú piccolo di 1×10^{-99}

12	**Line not found** (Linea non esistente): Sono state usate le istruzioni GOSUB, GOTO o THEN con riferimento ad una linea che non esiste.
13	**No Matching FOR Statement** (Errore NEXT senza FOR): L'istruzione NEXT è stata usata senza il corrispondente FOR oppure un'espressione nidificata FOR/NEXT è stata usata senza un adeguato accoppiamento (il messaggio d'errore viene riportato all'istruzione NEXT e non alla FOR).
14	**Line Length Error** (Errore di Lunghezza di Linea): L'espressione è troppo complessa o troppo lunga per essere svolta dal BASIC.
15	**GOSUB or FOR Line Deleted** (Le linee di GOSUB o FOR sono state cancellate): Il programma ha trovato l'istruzione RETURN o NEXT ma i corrispondenti GOSUB o FOR sono stati cancellati dopo l'ultimo comando RUN.
16	**RETURN Error** (Errore nell'Istruzione RETURN): Un RETURN è stato trovato senza il corrispondente GOSUB.
17	**SINTAX ERROR** (Errore di Sintassi): Il computer ha incontrato una linea contenente un illecito di sintassi.
18	**Invalid String Character** (Carattere di stringa non valido): La stringa nell'istruzione VAL non è una stringa numerica.

Note: I seguenti errori sono riferiti all'INPUT/OUTPUT (I/O) durante l'uso di dischetti, stampati ed altre unità periferiche. Ulteriori informazioni sono fornite con l'unità periferica.

19	**Load Program Too Long** (Il programma che si sta caricando con LOAD è troppo lungo): Non c'è memoria a sufficienza per completare l'istruzione LOAD.
20	**Device Number Error** (Errore di Numero del Dispositivo): Il numero del dispositivo è piú grande di 7 o uguale a 0.

21	**Load File Error** (Errore di Caricamento di un archivio [file]): L'utente ha fatto il tentativo di caricare un archivio non caricabile, o un archivio non etichettato dal BASIC. Per creare degli archivi etichettati usare il comando SAVE.
128	**Break Abort** (Aborto del tasto BREAK): L'utente ha premuto il tasto Break durante una operazione di I/O.
129	**IOCB[1] Already Open** (IOCB già aperto): Il blocco di controllo Input/Output è già aperto.
130	**Non existent Device** (Dispositivo non esistente): L'utente ha fatto il tentativo di accendere ad un dispositivo non identificato (ossia un dispositivo non contenuto nella tavola di distribuzione).
131	**IOCB[1] Write Only Error** (Errore IOCB di sola scrittura): Un comando READ è stato dato ad un dispositivo di sola scrittura (la stampante per esempio).
132	**Invalid Command** (Comando Nullo): Il comando non è idoneo per questo dispositivo.
133	**Device or File Not Open** (L'archivio [file] del Dispositivo non è Aperto): Nessun comando OPEN è stato dato al dispositivo.
134	**Bad IOCB[1] Number** (Numero di IOCB Illecito): Un comando WRITE è stato dato ad un dispositivo di sola lettura.
136	**E.O.F.** (E.O.F.): Il computer ha raggiunto la fine dell'archivio (file).
137	**Truncated Record** (Registrazione Troncata): Questo errore si verifica normalmente quando la registrazione letta è piú grande della massima dimensione della registrazione specificata nella chiamata al CIO (la massima dimensione di registrazione nel BASIC è di 119 bytes).
138	**Device Time-Out** (Time-out del Dispositivo): Il dispositivo non risponde.
139	**Device NAK** (Dispositivo NAK): I problemi si trovano nel connettore seriale o nell'unità periferica.

140 **Serial Bus Input Framing Error** (Errore nella Scrittura d'Ingresso sul Bus Seriale): L'informazione si è persa tra l'unità periferica ed il computer.

141 **Cursor Out of Range** (Cursore Fuori Campo): Il cursore è fuori campo per un modo particolare.

142 **Serial Bus Data Frame Overrun** (Traboccamento della Struttura di Dati sul Bus Seriale): L'informazione si è persa tra l'unità periferica ed il computer.

143 **Serial Bus Data Frame Overrun Error** (Errore nella Somma Verifica nella Struttura Dati del Bus Seriale): L'informazione si è persa tra l'unità periferica ed il computer.

144 **Device Done Error** (Errore Causato dal Dispositivo): L'utente ha fatto il tentativo di scrivere su un dischetto protetto.

145 **Read After Write Compare Error** (Errore di confronto nella lettura dopo la scrittura): L'utente ha fatto il tentativo di aprire lo schermo dell'Editore con un numero di modo grafico illecito.

146 **Function Not Implemented** (Funzione non eseguita): La funzione non è stata inclusa nella tavola di distribuzione.

147 **Insufficient RAM** (RAM Insufficiente): Non c'è abbastanza memoria di RAM per lavorare nel modo grafico selezionato.

160 **Drive Number Error** (Errore nel Numero dell'Unità a Disco): L'utente ha specificato il numero errato dell'unità a disco.

161 **Too Many OPEN Files** (Troppi Archivi [file] OPEN): Non c'è piú memoria di transito disponibile.

162 **Disk Full** (Disco Pieno): Non ci sono settori (spazi) disponibili sul dischetto.

163 **Unrecoverable System I/O Error** (Errore Irrecuperabile nel Sistema I/O): La versione DOS del disco potrebbe essere stata danneggiata.

164 **File Number Mismatch** (Disparità nel Numero d'Archivio): Il disco d'archivio potrebbe essere stato danneggiato.

165 **File Name Error** (Errore nel Titolo d'Archivio): Le specifiche d'archivio contengono dei caratteri illeciti.

166 **POINT Data Length Error** (Errore nella Lunghezza Dati di POINT): Il secondo parametro dell'istruzione POINT è troppo grande.

167 **File Locked** (Archivio Chiuso): L'utente ha tentato l'accesso ad un archivio chiuso, con scopi diversi che quello di leggerlo.

168 **Invalid Command** (Comando Nullo): Il comando dato in un codice operativo speciale è nullo.

169 **Directory Full** (Elenco Pieno): L'utente ha utilizzato tutti gli spazi disponibili (64 nomi d'archivio) nell'elenco.

170 **File Not Found** (Archivio non Trovato): L'utente ha tentato l'accesso ad un archivio che non esiste nell'elenco del dischetto.

171 **Invalid Point** (POINT Nullo): L'utente ha cercato di usare l'istruzione POINT su di un byte in un archivio che non è stato aperto per aggiornamento.

172 **Illegal Append** (Aggiunta Illecita): L'utente ha cercato di usare il DOS II per aprire ed aggiungere dati in un archivio DOS I. Il DOS II non si usa con gli archivi in DOS I. Usando il DOS II copia l'archivio di DOS I su un dischetto di DOS II.

173 **Bad Sectors at Format Time** (Settori Guasti Durante la Formattazione): L'unità a disco ha trovato dei settori guasti durante la formattazione di un dischetto. Usate un altro dischetto, poiché un dischetto con settori guasti non può essere formattato. Se questo errore persiste con diversi dischetti, l'unità a disco potrebbe aver bisogno di essere riparata.

[1] IOCB si riferisce al Blocco di Controllo I/O. Il numero del dispositivo è lo stesso di quello del IOCB.

E. Ricerca Guasti

D. Ho collegato il mio computer seguendo le istruzione del capitolo: Fare i collegamenti, ma quando lo accendo non succede niente. Cosa devo fare?

R. Se la parola READY non appare sullo schermo, rivedete i collegamenti. Assicuratevi che tutti i cavi e i fili siano ben inseriti e che la corrente arrivi al sistema. Provate ad aggiustare la sintonia del vostro televisore. Se non avete ancora ottenuto l'immagine giusta o le parole MEMORI TEST, il vostro computer potrebbe aver bisogno di assistenza. Per trovare il Centro Assistenza piú vicino, rivolgetevi al vostro negoziante o contattate la sede ATARI.

D. Cosa posso fare se il computer è acceso ma l'immagine TV è distorta?

R. Ci sono diverse cose da controllare:

- Assicuratevi che il cavo sia ben inserito nella scatola di commutazione TV e che l'interruttore sia nella posizione COMPUTER o GAME.

- Il TV deve ricevere un segnare da 300-OHM (300) dalla scatola di commutazione TV per funzionare bene. Controllate di aver seguito alla lettera le istruzioni della sezione: Collegamento del computer alla scatola di commutazione TV.

- Assicuratevi che l'interruttore del canale sul computer sia sul medesimo numero di canale del vostro TV.

- Provate un'altra cartuccia nell'apposito allocamento per determinare se il software funziona. Assicuratevi che la cartuccia sia saldamente inserita.

D. Quando premo il tasto [Help] non succede niente. Perché?

R. Il tasto [Help] funziona solo con alcuni programmi specifici. Dà accesso ad istruzioni supplementari quando ne avete bisogno. Se il computer non risponde, è probabile che il programma che state usando non sia stato predisposto per l'uso del tasto [Help].

D. Ho provato ad usare il BASIC ATARI incorporato nel computer. Ho battuto sulla tastiera un programma ma non gira. Perché?

R. Assicuratevi di aver premuto [Return] dopo ogni istruzione di programma, prima di battere RUN. Infatti premendo [Return], dite al computer che state per introdurre delle informazioni.

Un altro errore molto comune è quello di confondere lo zero (0) con la lettera maiuscola o (O). In effetti sembrano uguali, ma il computer li tratta in modo differente.

F. Risorse

Non è necessario che l'utente del computer ATARI debba lavorare o giocare in isolamento. La maggior parte dei problemi che incontrerà sono già stati risolti da altri. Questa esperienza è stata in gran parte documentata cosí da darvi un buon supporto. Le risorse elencate qui di seguito sono solo una parte dell'assistenza che vi può condurre verso nuove ed interessanti scoperte nell'uso di un computer. Potrete trovare quanto disponibile visitando le librerie specializzate o il vostro negoziante ATARI.

GRUPPI UTENTI

Potrete confrontare le informazioni che otterrete con altri proprietari dei Computer ATARI iscrivendovi ad un gruppo utenti ATARI. I gruppi utenti hanno delle riunioni mensili e pubblicano dei bollettini. Se avete bisogno d'aiuto per programmare in altri linguaggi di computer, o di consigli per acquistare del nuovo software, o di informazioni sugli ultimi prodotti per il vostro ATARI 130XE, li potrete trovare presso il gruppo utenti a voi piú vicino.

LIBRI

Un'ampia serie di libri è disponibile per gli utenti di ATARI. Per informazioni, contattate il vostro negoziante.

RELAZIONI CON I CLIENTI

Per ogni domanda o problemi riguardo i prodotti ATARI, scrivete ad ATARI ITALIA S.p.A., Via dei Lavoratori 19, 20092 Cinisello Balsamo – Milano.

G. Caratteristiche dell'ATARI 130XE

Processore:	Microprocessore 6502C con clock da 1.79 MHz
Chips su misura:	Chip GTIA: visualizzazione grafica Chip POKEY: generatore e controllo di suono Chip ANTIC: Input/Output per lo schermo e porte Chip FREDDY: sistema controllo di memoria
Memoria:	131,072 bytes RAM 24,576 bytes ROM (sistema operativo + linguaggio programmazione Basic)
Display:	11 modi grafici 256 colori (128 visualizzati ogni volta) Massima risoluzione grafica 320 × 192 Visualizzazione del testo: 40 colonne per 24 linee 5 modi di testo
Suono:	4 canali sonori indipendenti su una gamma di 3 1/2 ottave
Tastiera:	Completa con 62 tasti, tasto Help e quattro tasti per funzioni speciali; caratteri internazionali; 29 tasti grafici
Capacità di Programmazione	Linguaggio di programmazione ATARI Basic incorporato. Software compatibile con i computers ATARI 65XE e 800XL; unità a disco RAM con ATARI Dos 2.5
Ingresso/Uscita:	Alloggiamento per cartucce di Software Uscita per televisore Uscita per monitor 2 porte per dispositivi comandi giochi Connettore seriale per ingresso/uscita (I/O)
Alimentazione:	1 Amp. a 5V DC

H. Accedere alla potenzialità del 130XE

Il 130XE contiene 131,072 bytes Random Access Memory (RAM): ha cioè una potenza due volte superiore al 65XE ed all'800XL.
I programmi possono utilizzare questi bytes di memoria per immagazzinare un maggiore numero di dati. Se collegate un'unità a disco ATARI 1050 con l'ATARI Dos 2.5 potete utilizzare la memoria aggiuntiva come un RAMDISK, cioè un'unità a disco molto veloce (consultate il manuale relativo per ulteriori informazioni).
In ogni caso potete accedere all'uso dei 65,536 bytes RAM con l'ATARI Basic tramite il metodo "bank switching" (sistema di cambio di blocchi di memoria).
Quando viene usato il metodo bank switching il computer è costretto ad usare una maggiore memoria. Il bank switching spegne un settore di 16K RAM e ne usa un altro al suo posto.
Nel 130XE il secondo banco di memoria si trova nella mappa di memoria in locazione da 16384 a 32767 (in esadecimali, i numeri $4000 a $7FFF). Cambiando il bank switching, il computer seleziona il banco di memoria che occuperà l'area dei 16K.
Il bank switching si trova in memoria nella locazione 54017. Questa locazione è usata come l'indirizzo della porta B del chip 6520, adattatore di interfaccia per periferiche, che controlla l'input e l'output del computer.
Metà della locazione di memoria veniva usata dal computer precedentemente. I bits 4 e 5 selezionano il processore del computer che può accedere al banco aggiuntivo. Solitamente i bits sono fissati a 1 (il bit 4 blocca il banco del CPU e il bit 5 blocca il banco video). Cambiando un bit a 0, il processore 6502 permette l'accesso alla RAM aggiuntiva, mentre l'altro bit dà l'accesso al processore video ANTIC. Il cambio di entrambi i bits fa usare ai processori la memoria aggiuntiva. I bit 2 e 3 controllano il settore di memoria aggiuntiva che viene usato. Ci sono 4 possibili combinazioni dei 2 bits e 4 diversi settori di 16K di memoria tra i quali scegliere.
Di solito la locazione di memoria 54017 contiene 193. Usando il comando POKE del linguaggio Basic potete modificare la locazione di memoria per accedere alla RAM aggiuntiva. Per esempio, POKE 54017,225 seleziona il primo settore di RAM aggiuntiva e rende disponibile questo settore al chip 6502 (non al chip processore video ANTIC).
La formula per selezionare un banco di memoria nel 130XE è:
POKE 54017,193 + 4*ADDRESS + 16* MODE

INDICE DEI VALORI

INDIRIZZO	SETTORE DI MEMORIA
0	0 to 16383
1	16384 to 32767
2	32768 to 49151
3	49152 to 65535

MODO	ACCESSO 6502	ACCESSO ANTIC
0	AGGIUNTIVO	SI
1	NORMALE	SI
2	AGGIUNTIVO	NORMALE
3	NORMALE	NORMALE

LOCAZIONE DI MEMORIA 54017

Bit#	7	6	5	4	3	2	1	0
Valore	128	64	32	16	8	4	2	1
(se fissato)	1	1	?	?	?	?	×	1

Banco Video	Banco CPU	Banco MSB	Banco LSB
Disattiva VBE	Disattiva CBE		

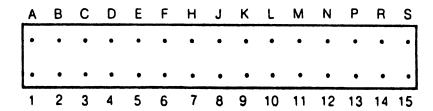
I bits 0, 6, 7 e 1 devono sempre essere fissati.

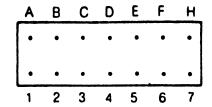
Se si fissa VBE o CBE, l'accesso corrispondente è normale. Se è 0, qualsiasi accesso all'indirizzo $4000–$7FFF va all'installazione aggiuntiva per mezzo del valore del banco bits.

I. Dettagli sul collegamento

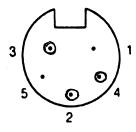
Porta seriale I/O (periferica)

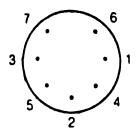
1. Clock Input
2. Clock Output
3. Data Input
4. Campo
5. Data Output
6. Campo
7. Comando
8. Controllo motore
9. Procedere
10. + 5/Ready
11. Audio Input
12. + 12V
13. Interruttore

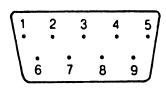
Alloccamento cartuccia

1.	S4	A.	RD4
2.	A3	B.	GND
3.	A2	C.	A4
4.	A1	D.	A5
5.	A0	E.	A6
6.	D4	F.	A7
7.	D5	H.	A8
8.	D2	J.	A9
9.	D1	K.	A12
10.	D0	L.	D3
11.	D6	M.	D7
12.	S5	N.	A11
13.	+5V	P.	A10
14.	RD5	R.	R/W
15.	CCTL	S.	B02

INTERFACCIA POTENZIATA PER CARTUCCIA (ECI)

1. EXSEL
2. RST
3. D1XX
4. MPD
5. Audio
6. REF
7. +5V

A. Reserved
B. IRQ
C. HALT
D. A13
E. A14
F. A15
H. GND

Connettore monitor

1. Composite Luminance
2. Ground
3. Audio Output
4. Composite Video
5. Composite Chroma

PRESA ADATTATORE CORRENTE

1. +5V
2. Shield
3. Ground
4. +5V
5. Ground
6. +5V
7. Ground

PORTA DI CONTROLLO

1. (Joystick) Input anteriore
2. (Joystick) Input posteriore
3. (Joystick) Input sinistro
4. (Joystick) Input destro
5. Input B potenziometro
6. Trigger Input
7. +5V
8. Campo
9. Input A potenziometro

INDICE ALFABETICO

D

E

J

L

M